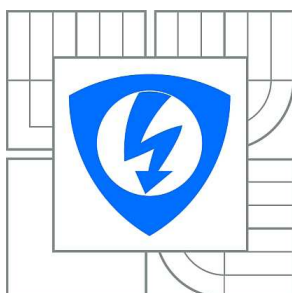


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

PRŮMYSLOVÝ REGULÁTOR PID S AUTOTUNEREM A VIZUALIZACÍ

INDUSTRIAL PID CONTROLLER WITH AUTOTUNING AND VISUALISATION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

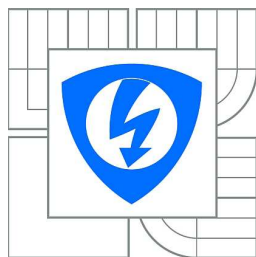
Bc. PAVEL VÁVRA

VEDOUCÍ PRÁCE

SUPERVISOR

prof. Ing. PETR PIVOŇKA, CSc.

BRNO 2010



**VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky
a komunikačních technologií**

Ústav automatizace a měřicí techniky

Diplomová práce

magisterský navazující studijní obor
Kybernetika, automatizace a měření

Student: Bc. Pavel Vávra

ID: 78334

Ročník: 2

Akademický rok: 2009/2010

NÁZEV TÉMATU:

Průmyslový regulátor PID s autotunerem a vizualizací

POKYNY PRO VYPRACOVÁNÍ:

Vytvořte průmyslově použitelný regulační člen a implementujte jej do zařízení Power Panel 282 firmy B&R, který integruje programovatelný automat a operátorský panel s dotykovou obrazovkou. Regulační člen bude umožňovat změnu struktury regulátoru, možnost beznárazového přepnutí na ruční režim a mezi řídicími algoritmy. Regulační člen bude ovládán přes dotykovou obrazovku a bude mít možnost adaptace parametrů pomocí klasických ident. metod a rozkmitáváním. Porovnejte a zhodnoťte vlastnosti. Porovnejte s standardním adaptivním regulátorem firmy B&R.

DOPORUČENÁ LITERATURA:

PIVOŇKA, P.: Optimalizace regulátorů. VUT Brno, skriptum, 2005.

PIVOŇKA, P.: Číslicová řídicí technika, VUT Brno, skriptum, 2003

BOBÁL, V. a kol.: Praktické aspekty samočinně se nastavujících regulátorů. VUTUM, Brno, 1999.

Termín zadání: 8.2.2010

Termín odevzdání: 24.5.2010

Vedoucí práce: prof. Ing. Petr Pivoňka, CSc.

prof. Ing. Pavel Jura, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Tato diplomová práce se zabývá implementací průmyslově použitelného regulačního členu do zařízení Power Panel firmy B&R, v programovém prostředí Automation Studio 3.0.71. Jedná se o PID regulátor s autotunerem a vizualizací, který umožňuje beznárazové přepínání mezi řídicími algoritmy a ručním režimem. Byl implementován diskrétní PID a I-PD regulátor s filtrací derivační složky a dynamickým antiwindupem. Parametry PID regulátoru je možné nastavit pomocí kritického zesílení a kritické periody kmitů podle pravidel Zieglera-Nicholse. Kritické parametry řízené soustavy je možné získat ze dvou identifikačních metod, rekurzivní metody nejmenších čtverců a relé ve zpětné vazbě. Rekurzivní metoda nejmenších čtverců byla implementována se směrovým zapomínáním. Při implementaci relé ve zpětné vazbě byly použity dva typy relé: ideální a saturační pro zlepšení přesnosti hledaných kritických hodnot. Celé řešení je naprogramováno v jazyce ANSI C, který Automation Studio podporuje. Vytvořený regulátor je ovládán pomocí dotykové obrazovky, která je do Power Panelu integrována. Na obrazovce jsou dále zobrazovány průběhy procesních veličin. Pro srovnání byl implementován adaptivní regulátor firmy B&R, který je standardně dodáván s Automation Studií v knihovně LoopConR. Všechny vytvořené algoritmy byly ověřovány nejprve na matematickém modelu soustavy a poté na reálném modelu v laboratoři. První část práce se zabývá teoretickým rozбором metod, které byly použity. Vlastní praktická realizace je popsána v druhé části této diplomové práce.

Klíčová slova

PID regulátor, adaptivní regulátor, beznárazové přepínání, identifikace, rekurzivní metoda nejmenších čtverců, relé ve zpětné vazbě, Power Panel, Automation Studio, ANSI C, LoopConR

Abstract

This thesis deals with implementation of industrial used controlling block into Power Panel equipment of B&R in integrated development environment Automation Studio 3.0.71. It is PID controller with autotuner and visualization, which allowed bumpless transfer among controlling algorithms and manual control. PID and I-PD controllers with filtering of derivative action and dynamic antiwindup were implemented. Parameters of PID controller is possible to tune with the aid of ultimate gain and ultimate period according rules of Ziegler-Nichols. Ultimate parameters of controlled plant is possible to acquire with the aid of two identification methods, recursive least squares method and relay feedback. Recursive least squares method was implemented with directional forgetting. For relay feedback were used two types of relays: ideal and saturation relay for improving accuracy of searched ultimate values. The whole solution is programmed in ANSI C which Automation Studio supports. Created controller is control with the aid of touchscreen which is integrated in Power Panel. Trends of process values are viewed on the screen too. For comparison adaptive controller by B&R was implemented. This controller is standardly supplied with Automation Studio in LoopConR library. All created algorithms were first validated on mathematical model of plant and then on real model in laboratory. The first part of thesis deals with theoretic analysis of used methods. The practical realization is described in the second part of this diploma thesis.

Keywords

PID controller, adaptive controller, bumpless transfer, identification, recursive least squares method, relay feedback, Power Panel, Automation Studio, ANSI C, LoopConR

Bibliografická citace

VÁVRA, P. *Průmyslový regulátor PID s autotunerem a vizualizací* . Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 68 s. Vedoucí diplomové práce prof. Ing. Petr Pivoňka, CSc.

Prohlášení

„Prohlašuji, že svou diplomovou práci na téma Průmyslový regulátor PID s autotunerem a vizualizací jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.“

V Brně dne: **24. května 2010**

.....

podpis autora

Poděkování

Děkuji vedoucímu diplomové práce prof. Ing. Petru Pivoňkovi, CSc. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne: **24. května 2010**

.....

podpis autora

1. ÚVOD	8
2. IDENTIFIKACE KRITICKÝCH PARAMETRŮ REGULOVANÉ SOUSTAVY	10
2.1 Průběžná metoda nejmenších čtverců	10
2.1.1 Směrové exponenciální zapomínání	11
2.1.2 Řád modelu a počáteční nastavení algoritmu	13
2.2 Výpočet kritických parametrů z parametrů modelu.....	14
2.2.1 Výpočet kritického zesílení pro model prvního řádu	15
2.2.2 Výpočet kritického zesílení pro model druhého řádu.....	16
2.2.3 Výpočet kritického zesílení pro model třetího řádu	16
2.3 Identifikace kritických parametrů pomocí relé ve zpětné vazbě.....	19
2.3.1 Saturační relé ve zpětné vazbě.....	20
2.3.2 Možný problém při použití saturačního relé.....	23
3. ŘÍDÍCÍ ALGORITMY	25
3.1 Diskrétní PID regulátor s filtrací derivační složky a antiwindupem.....	25
3.2 Omezení překmitu regulované veličiny	27
3.3 Beznárazové Přepínání.....	28
4. STRUKTUROVANÝ NÁVRH SOFTWARE	30
4.1 Blok Ident.....	33
4.1.1 Rekursivní metoda nejmenších čtverců a návrh parametrů.....	34
4.1.2 Identifikace pomocí relé ve zpětné vazbě.....	34
4.1.3 Identifikace a řízení pomocí knihovny LoopConR	38
4.2 Blok Control.....	41
4.2.1 Beznárazové přepínání.....	42
5. PRAKTICKÉ ASPEKTY IMPLEMENTACE	44
5.1 Použitý hardware	44
5.1.1 Vstupně-výstupní karty.....	44
6. VIZUALIZACE A OVLÁDÁNÍ.....	49
7. OVĚŘENÍ A SIMULACE.....	53
8. ZÁVĚR.....	61

1. ÚVOD

Hlavním cílem diplomové práce byla implementace průmyslově použitelného PID regulátoru s autotunerem a vizualizací. Řídicí algoritmus regulátoru je realizován v diskrétní formě. To ale neznamená pouze nahradit integraci sumací a derivaci diferencí. Jsou nutné další úpravy, aby mohl být regulátor nasazen s úspěchem na reálných systémech. V první řadě musí být zajištěno omezení přebuzení integrační složky regulátoru tzv. wind-up jevu. Neméně důležitá je správná diskrétní realizace derivační složky. Byl implementován přístup neměnné plochy odezvy derivační složky, navržený v [7]. Důležitou součástí regulátorů nasazených v průmyslu je možnost beznárazového přepínání mezi řídicími algoritmy a ručním režimem.

Jelikož jsou PID regulátory masivně nasazovány v praxi a jejich ruční seřízení může být časově náročné, je snaha doplnit je o možnost automatického nastavení jejich parametrů. Byly použity dva přístupy. Rekurzivní metoda nejmenších čtverců, jejíž podstatou je minimalizace kvadrátu rozdílu odhadovaného a měřeného výstupu. Metoda byla implementována se směrovým zapomínáním, který odstraňuje tzv. „estimator windup“. Identifikace je dále možná pomocí relé ve zpětné vazbě. Reléovou zpětnou vazbou lze dosáhnout rozkmitání regulované veličiny kritickou periodou kmitů. Z průběhu regulované veličiny lze také zjistit hodnotu kritického zesílení. Ideální relé bylo doplněno o saturační relé, které přispělo ke zpřesnění hledaných kritických parametrů [6].

Navržené řešení bude implementováno v programovém prostředí Automation Studio 3.0.71, v jazyce ANSI C. Pro porovnání vytvořeného regulátoru bude také implementován adaptivní regulátor od firmy B&R, který je obsažen v knihovně LoopConR. Knihovna je součástí Automation Studia. Tento adaptivní regulátor lze nastavit jako P, PI nebo PID s možností volby dalších parametrů. Identifikace řízené soustavy je možná pomocí relé ve zpětné vazbě nebo z odezvy na jednotkový skok.

Funkčnost vytvořeného regulačního členu bude nejprve ověřována v simulátoru AR000, který představuje virtuální PLC v počítači. Jako regulovanou soustavu lze použít matematický model v simulinku nebo funkční blok LCRPT2

z knihovny LoopConR, který představuje setrvačný členek druhého řádu. Model v simulinku má tu výhodu, že se v něm dá simulovat i šum. Propojení simulinku a virtuálního PLC bude realizováno pomocí bloku mk_pvi. Dále bude regulátor testován na Power Panelu. K Power Panelu bude pomocí analogového vstupu a výstupu připojen reálný model.

Práce je členěna následovně: druhá kapitola se zabývá rozбором identifikačních metod a to rekurzivní metodou nejmenších čtverců se směrovým zapomínáním a identifikací pomocí relé ve zpětné vazbě. Třetí kapitola teoreticky popisuje použité řídicí algoritmy. Ve čtvrté kapitole začíná praktická část práce popisem strukturovaného návrhu softwaru. Pátá kapitola se věnuje použitému hardwaru. Šestá kapitola popisuje stručně ovládání a vizualizaci a sedmá prezentuje výsledky simulací na reálném modelu. Shrnutí práce se nachází v osmé kapitole. Detailnější popis vytvořeného regulátoru se nachází v příloze.

2. IDENTIFIKACE KRITICKÝCH PARAMETRŮ REGULOVANÉ SOUSTAVY

V navrženém regulačním členu je identifikace kritických parametrů možná pomocí dvou přístupů: rekurzivní metody nejmenších čtverců a relé ve zpětné vazbě. Výstupem rekurzivní metody nejmenších čtverců jsou parametry modelu. Model může být prvního, druhého nebo třetího řádu. Kritické parametry užitě pro návrh regulátoru podle ZN, je poté ještě potřeba spočítat. Naproti tomu při identifikaci pomocí relé ve zpětné vazbě jsou výstupem už přímo kritická perioda kmitů a kritické zesílení.

2.1 PRŮBĚŽNÁ METODA NEJMENŠÍCH ČTVERCŮ

Princip metody nejmenších čtverců spočívá v následující myšlence. Algoritmus výpočtu se snaží nastavit parametry zvoleného modelu tak, aby rozdíl mezi měřenou výstupní veličinou z procesu y a odhadovanou výstupní veličinou z modelu \hat{y} byl co nejmenší (Obr. 2.1).

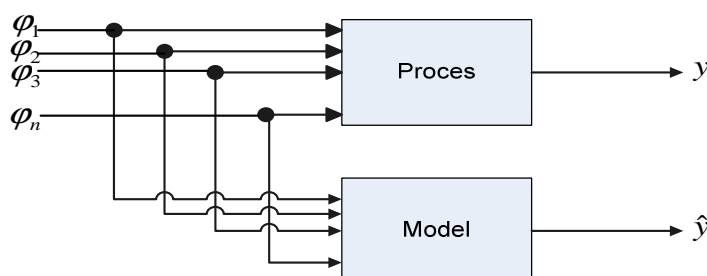
Výstup z modelu \hat{y} se počítá podle následujícího regresního vztahu:

$$\hat{y}(i) = \varphi_1(i) \cdot \theta_1 + \varphi_2(i) \cdot \theta_2 + \dots + \varphi_n(i) \cdot \theta_n = \varphi^T(i) \cdot \theta + \varepsilon \quad (2.1)$$

kde φ je vektor měřených minulých vstupů a výstupů

θ je odhadovaný vektor parametrů modelu

ε je difference mezi výstupem z procesu y a výstupem z modelu \hat{y}



Obr. 2.1: Souvislost mezi modelem a procesem [1]

Po n -měřeních je již určen první odhad parametrů modelu θ . Pokud se vektor θ během trvání identifikace nemění, probíhá výpočet podle následující rovnice:

$$\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} \varphi_{11} & \cdots & \varphi_{1n} \\ \vdots & \ddots & \vdots \\ \varphi_{n1} & \cdots & \varphi_{nn} \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix} \quad (2.2)$$

Předchozí vztah zapíšeme maticově a vyjádříme z něho chybu výpočtu. Tato chyba se bude později minimalizovat.

$$y = \Phi \theta + \varepsilon \Rightarrow \varepsilon = y - \Phi \theta \quad (2.3)$$

Jelikož může být chyba z modelu oproti výstupu z procesu kladná i záporná, metoda minimalizuje kvadrát difference ε . Proto hledáme minimum následující funkce:

$$J(\theta) = \frac{1}{2} \varepsilon^T \varepsilon \quad (2.4)$$

Minimum získáme, pokud do předchozího vztahu dosadíme rovnici 2.3 a derivujeme podle hledaného vektoru parametrů θ . Po úpravě získáváme vztah pro výpočet parametrů regresního modelu:

$$\theta = (\Phi^T \Phi)^{-1} \Phi^T y \quad (2.5)$$

Popsaným způsobem by se v každém kroku odhadu zvětšovala matice minulých vstupů a výstupů Φ . Proto je zaveden rekurzivní výpočet, který v každém kroku zahrnuje nově naměřené hodnoty do výpočtu.

2.1.1 Směrové exponenciální zapomínání

Parametry řízeného procesu se mohou vlivem stárnutí během identifikace měnit. Proto se do rekurzivního výpočtu zavádí ještě exponenciálního zapomínání, které přiřazuje různě starým datům různou váhu. Klasická metoda nejmenších čtverců s exponenciálním zapomínáním však pracuje dobře jen tehdy pokud je systém dostatečně buzen během celé doby identifikace. Pokud je systém v ustáleném stavu, dochází k tzv. „estimator windup“ jevu [2]. V klasické metodě nejmenších čtverců se totiž v každém kroku aktualizuje kovarianční matice podle vztahu:

$$P(k) = (P(k-1) - K(k)\varphi^T(k)P(k-1)) / \lambda \quad (2.6)$$

Pokud budeme uvažovat případ, že systém není buzen vůbec ($\varphi = 0$), redukuje se rovnice 2.6 následovně:

$$P(k) = \frac{1}{\lambda} P(k-1) \quad (2.7)$$

Jelikož se koeficient exponenciálního zapomínání volí v rozmezí $0 < \lambda < 1$, vyplývá z tohoto vztahu, že prvky matice P rostou s časem exponenciálně. K tomuto jevu dochází, i když regresní vektor φ zůstává konstantní. K eliminaci tohoto nedostatku lze použít několik přístupů, např. směrové zapomínání navržené v [3].

Jádrem algoritmu je následující rozložení informační matice R (inverzní matice ke kovarianční matici P).

$$R(t-1) = R_1(t-1) + R_2(t-1) \quad (2.8)$$

Pro matici R_1 platí následující:

$$R_1(t-1)\varphi(t) = 0, \quad \varphi(t) \neq 0 \quad (2.9)$$

Matice R_2 potom představuje část informační matice R , která by měla být zapomínána. Pokud je systém dostatečně buzen, chová se algoritmus jako klasické exponenciální zapomínání. Při nedostatečném buzení nedochází k „estimator windupu“, protože žádná informace není zapomínána v části informační matice, která je ortogonální k prostoru vektoru φ . Celý algoritmus je popsán v následujících rovnicích:

$$\begin{aligned} P(t-1) &= P(t-1) + \frac{1-\lambda}{\lambda} \frac{\varphi(t)\varphi^T(t)}{\varphi^T(t)R(t-1)\varphi(t)}, \quad |\varphi(t)| > \varepsilon \\ P(t-1) &= P(t-1), \quad |\varphi(t)| \leq \varepsilon \\ K(t) &= \frac{P(t-1)\varphi(t)}{1 + \varphi^T(t)P(t-1)\varphi(t)} \\ \hat{\theta}(t) &= \hat{\theta}(t-1) + K(t)[y(t) - \varphi^T(t)\hat{\theta}(t-1)] \\ P(t) &= P(t-1) - \frac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{1 + \varphi^T(t)P(t-1)\varphi(t)} \\ M(t) &= (1-\lambda) \frac{R(t-1)\varphi(t)\varphi^T(t)}{\varphi^T(t)R(t-1)\varphi(t)}, \quad |\varphi(t)| > \varepsilon \\ M(t) &= 0, \quad |\varphi(t)| \leq \varepsilon \\ R &= [I - M(t)]R(t-1) + \varphi(t)\varphi^T(t) \end{aligned} \quad (2.10)$$

kde λ je koeficient exponenciálního zapomínání

R je informační matice

ε je střední hodnota šumu, tzn. určuje pásmo necitlivosti okolo nuly, kdy se hodnoty měřených vstupů a výstupů považují za nulu

K je vektor, který určuje jak moc velký bude mít vliv na nový odhad parametrů rozdíl mezi skutečným a odhadovaným výstupem

2.1.2 Řád modelu a počáteční nastavení algoritmu

Algoritmus rekurzivní metody nejmenších čtverců může identifikovat model v jakémkoli tvaru. Jelikož návrh parametrů regulátoru bude realizován pomocí modifikované metody ZN pro seřizování číslicových PID regulátorů, musí mít model určitou strukturu. Jak již bylo zmíněno na začátku kapitoly, model může být prvního, druhého nebo třetího řádu. Může mít proto jeden z následujících tvarů:

$$\begin{aligned} F_c(z) &= \frac{Y(z)}{U(z)} = \frac{b_1 z^{-1}}{1 + a_1 z^{-1}} \\ F_c(z) &= \frac{Y(z)}{U(z)} = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \\ F_c(z) &= \frac{Y(z)}{U(z)} = \frac{b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} \end{aligned} \quad (2.11)$$

Počáteční nastavení algoritmu je voleno následovně (I značí jednotkovou matici):

$$\begin{aligned} P_0 &= 10^9 I \\ R_0 &= \frac{1}{10^9} I \\ \theta_0 &= [1 \quad 0 \quad \dots \quad 0]^T \\ \lambda &= 0,995 \end{aligned} \quad (2.12)$$

Pokud bude mít vektor φ následující tvar (pro druhý řád),

$$\varphi^T = [u(i-1) \quad u(i-2) \quad -y(i-1) \quad -y(i-2)] \quad (2.13)$$

potom dostaneme vektor odhadovaných parametrů modelu v následujícím tvaru:

$$\theta = [\hat{b}_1 \quad \hat{b}_2 \quad \hat{a}_1 \quad \hat{a}_2] \quad (2.14)$$

Výsledky identifikace rekurzivní metodou nejmenších čtverců jsou silně závislé na periodě vzorkování. Delší perioda vzorkování působí pro vysokofrekvenční šum jako filtr.

2.2 VÝPOČET KRITICKÝCH PARAMETRŮ Z PARAMETRŮ MODELU

Kritické parametry se počítají bez rozkmitávání reálné soustavy, což je velká výhoda. Vstupem do výpočtu jsou parametry modelu z identifikace pomocí metody nejmenších čtverců a výstupem je kritická perioda a kritické zesílení. Tyto hodnoty jsou závislé na periodě vzorkování a počítají se z nich parametry regulátoru podle Zieglera-Nicholse.

Máme diskrétní přenosovou funkci regulované soustavy ve tvaru:

$$G_P(z) = \frac{Y(z)}{U(z)} = \frac{z^{-d} B(z^{-1})}{A(z^{-1})} \quad (2.15)$$

kde d je počet kroků dopravního zpoždění. Dále předpokládáme proporcionální regulátor ve tvaru:

$$G_R(z) = \frac{U(z)}{E(z)} = K_P \quad (2.16)$$

Přenos uzavřeného obvodu vyjadřuje následující vztah:

$$G_W(z) = \frac{Y(z)}{W(z)} = \frac{G_P(z)G_R(z)}{1 + G_P(z)G_R(z)} = \frac{z^{-d} K_P B(z^{-1})}{A(z^{-1}) + z^{-d} K_P B(z^{-1})} \quad (2.17)$$

Jmenovatel rovnice pro uzavřený obvod je:

$$D(z) = A(z^{-1}) + z^{-d} K_P B(z^{-1}) \quad (2.18)$$

což je charakteristický polynom. Póly tohoto polynomu určují dynamické chování celého systému. Aby byl regulační obvod na mezi stability, což potřebujeme k určení kritické periody a kritického zesílení, musí alespoň jeden pól ležet na jednotkové kružnici. Mohou nastat dva případy:

- charakteristická rovnice může mít dva komplexně sdružené póly $z_{1,2} = \alpha \pm j\beta$ a nějaké další póly
- charakteristický polynom má jeden nebo více reálných pólů v -1.

Na umístění pólů je závislý výpočet kritické periody kmitů. Základní definice Z-transformace je určena vztahem:

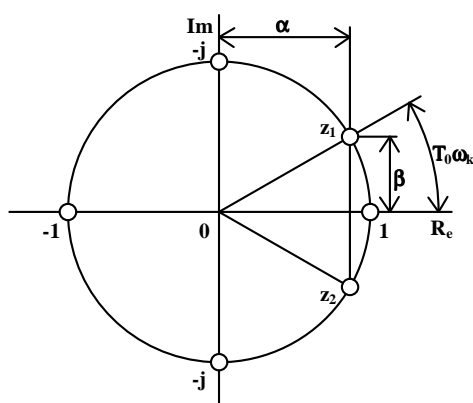
$$z = e^{j\omega T_0} = \cos \omega T_0 + j \sin \omega T_0 \quad (2.19)$$

Potom vzorce pro výpočet kritické periody kmitů jsou pro dva komplexně sdružené póly (Obr. 2.2):

$$\cos(\omega_K T_0) = \alpha \quad \omega_K = \frac{1}{T_0} \arccos \alpha \quad T_K(T_0) = \frac{2\pi}{\omega_K} \quad (2.20)$$

V případě reálných kritických pólů platí pro kritickou periodu kmitů následující vztahy:

$$\cos(\omega_K T_0) = -1 \quad \omega_K = \frac{\pi}{T_0} \quad T_K(T_0) = 2T_0 \quad (2.21)$$



Obr. 2.2: Umístění kritických pólů na jednotkové kružnici

2.2.1 Výpočet kritického zesílení pro model prvního řádu

Pokud budeme systém aproximovat prvním řádem a nebudeme uvažovat dopravní zpoždění, má charakteristický polynom (rovnice 2.18) tvar v kladných mocninách operátoru z :

$$D(z) = z + a_1 + K_p b_1 \quad (2.22)$$

U prvního řádu může existovat pouze jeden kritický reálný pól v -1 , proto se kritická perioda spočítá podle rovnice 2.21. Řešíme polynomiální rovnici, jejímž výsledkem je kritické zesílení:

$$z + a_1 + K_p b_1 = z + 1 \Rightarrow K_{PK}(T_0) = \frac{1 - a_1}{b_1} \quad (2.23)$$

2.2.2 Výpočet kritického zesílení pro model druhého řádu

V případě komplexně sdružených pólů, pokud neuvažujeme dopravní zpoždění a převedeme vše do kladných mocnin, dostáváme tvar charakteristické rovnice:

$$D(z) = z^2 + z(a_1 + K_{PK}(T_0)b_1) + a_2 + K_{PK}b_2 = (z - \alpha - j\beta)(z - \alpha + j\beta) = z^2 - 2\alpha z + 1 \quad (2.24)$$

Porovnáním koeficientů u stejných mocnin obdržíme vztahy pro výpočet K_{PK} a α (potřebná pro výpočet T_K - rovnice 2.20):

$$K_{PK}(T_0) = \frac{1 - a_2}{b_2} \quad \alpha = \frac{a_2b_1 - a_1b_2 - b_1}{2b_2} \quad (2.25)$$

V případě jednoho reálného pólu -1 a ostatních uvnitř jednotkové kružnici dostáváme:

$$D(z) = z^2 + z(a_1 + K_{PK}(T_0)b_1) + a_2 + K_{PK}b_2 = (z + 1)(z + f_1) = z^2 + z(1 + f_1) + f_1 \quad (2.26)$$

Stejným postupem jako v předešlém případě obdržíme vztah pro K_{PK} :

$$K_{PK}(T_0) = \frac{a_1 - a_2 - 1}{b_2 - b_1} \quad (2.27)$$

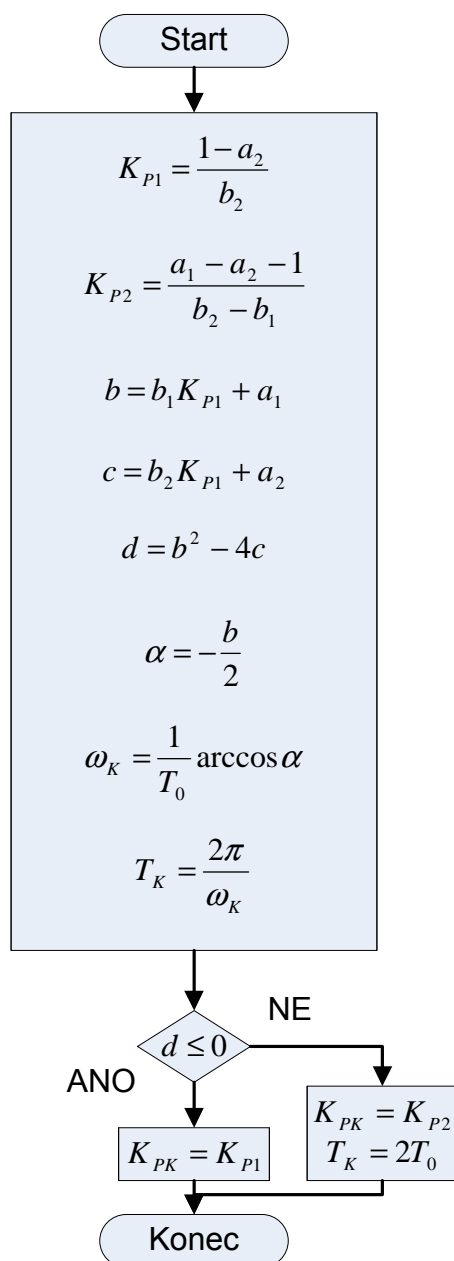
Jestli se bude kritické zesílení počítat podle rovnice 2.25, nebo 2.27 se rozhoduje podle diskriminantu charakteristického polynomu:

$$D(z) = z^2 + z(a_1 + K_{PK}(T_0)b_1) + a_2 + K_{PK}b_2 = 0 \quad (2.28)$$

Pokud je záporný (dva komplexně sdružené póly), počítá se podle vzorce 2.25, pokud je kladný, počítá se podle vzorce 2.27. Celý postup výpočtu je přehledně shrnut na Obr. 2.3.

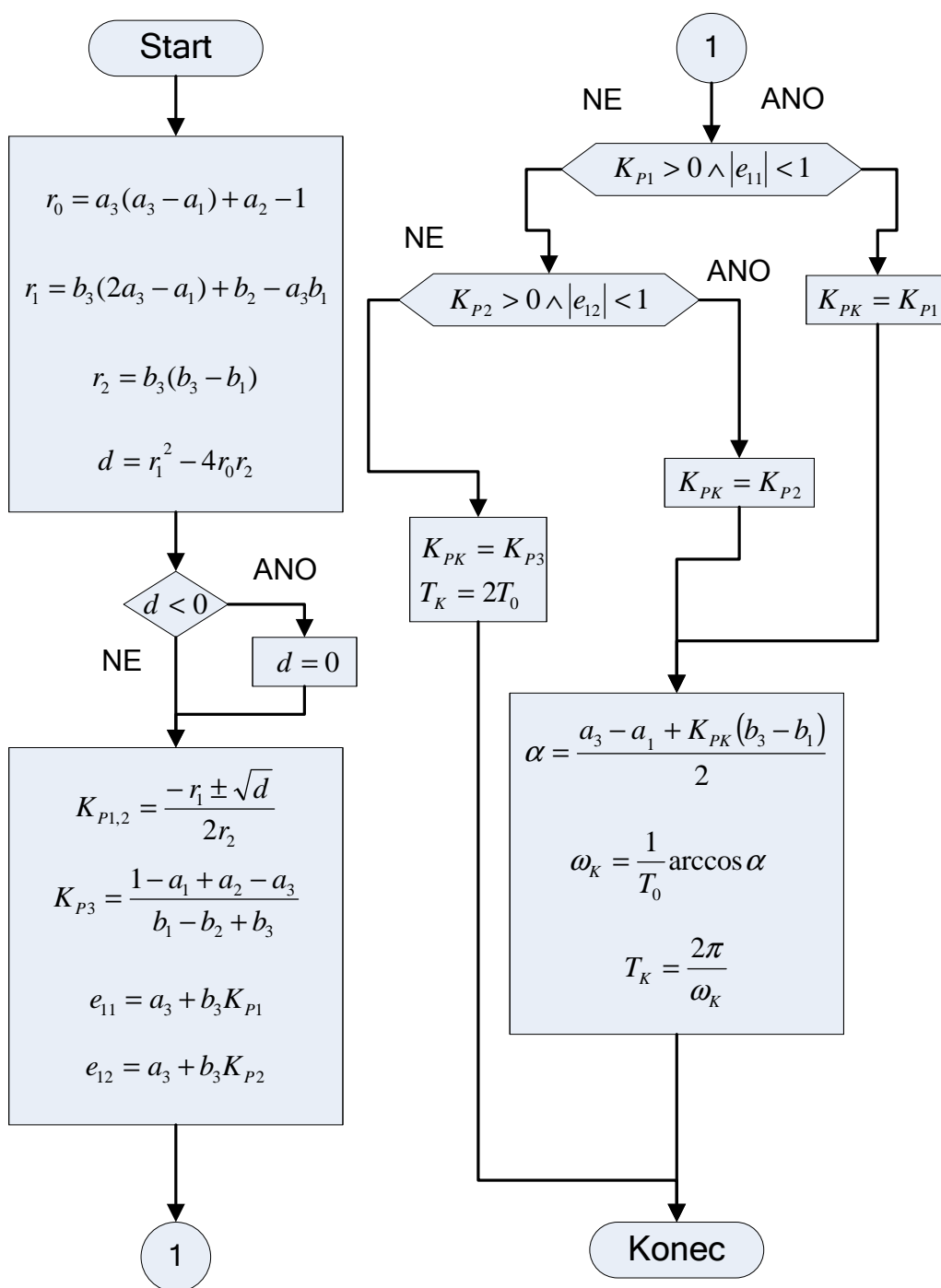
2.2.3 Výpočet kritického zesílení pro model třetího řádu

Postup výpočtu pro model třetího řádu je obdobný jako v předchozích dvou případech. Vyjádří se obecný tvar charakteristického polynomu a uvažují se případy



Obr. 2.3: Výpočet kritického zesílení a kritické periody kmitů pro druhý řád

s komplexně sdruženými póly a reálnými póly. Porovnají se koeficienty u stejných mocnin a vyjádří se vzorce pro výpočet kritického zesílení a kritické periody kmitů. Odvození lze nalézt v [4], odkud jsou i všechny obrázky z této kapitoly. Na Obr. 2.4 je postup výpočtu.

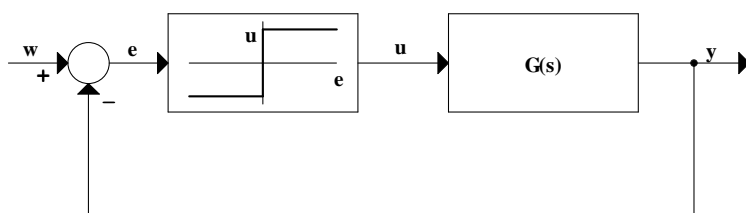


Obr. 2.4 Výpočet kritického zesílení a kritické periody kmitů pro třetí řád

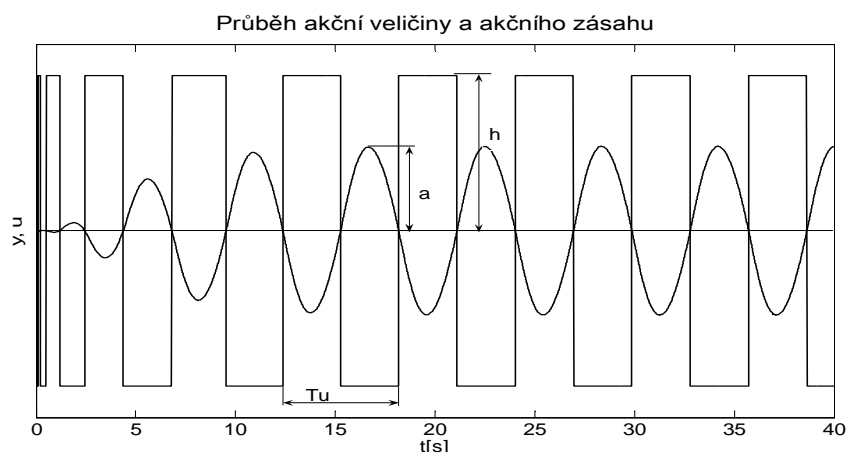
2.3 IDENTIFIKACE KRITICKÝCH PARAMETRŮ POMOCÍ RELÉ VE ZPĚTNÉ VAZBĚ

Výstupem identifikace kritických parametrů pomocí relé ve zpětné vazbě jsou již přímo kritická perioda a kritické zesílení. Metoda má tři hlavní výhody [5]. Zjišťuje parametry řízené soustavy při kritické frekvenci. Tyto parametry jsou poté využity při návrhu regulátoru. Za druhé experiment je prováděn v uzavřené smyčce, proto při správném výběru parametrů relé máme možnost systém držet v lineární oblasti, která nás zajímá. Za třetí, není nutná přesná volba odpovídající vzorkovací periody, ke které je nutná apriorní znalost řízené soustavy.

Zapojíme-li zpětnovazební regulační obvod podle Obr. 2.5, regulovaná veličina se rozkmitá kritickou periodou kmitů, protože fázový rozdíl mezi vstupním a výstupním signálem soustavy je π . Příklad průběhu akční a regulované veličiny při řízení pomocí relé je na Obr. 2.6.



Obr. 2.5 Zpětnovazební řízení pomocí relé



Obr. 2.6 Průběh regulované a akční veličiny při řízení pomocí relé ve zpětné vazbě

Pokud je systém typu dolní propust, můžeme po rozložení průběhu akčního zásahu ve Fourierovu řadu uvažovat pouze první harmonickou (vyšší harmonické řízená soustava odfiltruje). Z průběhu regulované veličiny určíme kritickou periodu kmitů a kritické zesílení, které se počítá podle následující rovnice.

$$K_u = \frac{4h}{\pi a} \quad (2.29)$$

kde h – amplituda relé

a – amplituda kmitů regulované veličiny

Je velmi důležité si uvědomit, že takto získané hodnoty kritických parametrů jsou pouze aproximací skutečných hodnot. Důvodem je uvažování pouze první harmonické Fourierova rozvoje akčního zásahu. Pokud chceme zpřesnit odhad, můžeme použít saturační relé [6].

2.3.1 Saturační relé ve zpětné vazbě

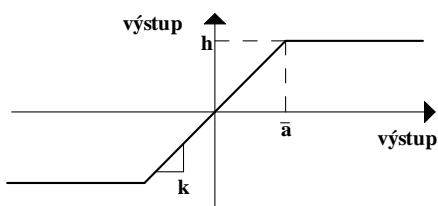
Hlavní příčina nepřesnosti odhadu kritických parametrů při použití ideálního relé pramení z náhrady obdélníkového průběhu první harmonickou Fourierova rozvoje (sinusový průběh). Použitím saturačního relé se snažíme obdélníkový průběh na výstupu z relé nahradit průběhem, který se bude více podobat sinusovému průběhu. Aproximace bude díky tomu zavádět do odhadu kritických parametrů menší chybu. Ideální relé se nahradí saturačním, které charakterizují dva parametry: výstupní hodnota při saturaci h a sklon k (Obr. 2.7). Pokud bude mít regulační odchylka e sinusový průběh,

$$e(t) = a \sin \omega t \quad (2.30)$$

bude mít výstup z relé (akční zásah u) hodnoty podle následující rovnice:

$$\begin{aligned} |e| \leq \bar{a} &\Rightarrow u = ke \\ e > \bar{a} &\Rightarrow u = h \\ e < -\bar{a} &\Rightarrow u = -h \end{aligned} \quad (2.31)$$

Tzn. že výstup bude mít sinusový průběh, jehož amplituda bude omezená hodnotou amplitudy relé h .



Obr. 2.7 Charakteristika saturačního relé

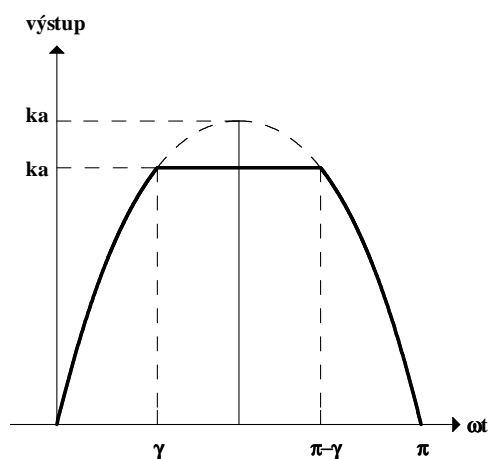
Jak moc bude výstup omezen lze charakterizovat pomocí úhlu γ :

$$\gamma = \sin^{-1}\left(\frac{\bar{a}}{a}\right) \quad (2.32)$$

kde a - amplituda odchylky (rovnice 2.30)

\bar{a} - maximální hodnota vstupu relé, při které je výstup přímo úměrný vstupu (Obr. 2.7)

Jelikož je výstup z relé periodický, můžeme pro analýzu vztahu 2.32 uvažovat pouze polovinu periody výstupu z relé. Pokud fáze leží mezi γ a $\pi - \gamma$, je výstup roven h . Naopak pro hodnoty $\omega t < \gamma$ a $\omega t > \pi - \gamma$ bude mít výstup sinusový průběh (Obr. 2.8). Hodnota úhlu γ nabývá hodnot z intervalu $<0; \pi/2>$. Krajní hodnota 0 představuje ideální relé ($\bar{a} = 0$). Pravá krajní hodnota představuje situaci, kdy relé na výstupu vůbec „neořezává“ ($\bar{a} = a$). Pokud vyjádříme výstup saturačního relé pomocí úhlu γ , dostaneme následující vztah (rovnice 2.33):



Obr. 2.8 Grafická reprezentace úhlu γ

$$\begin{aligned} u(t) &= \frac{h}{\sin \gamma} \sin \omega t & 0 \leq \omega t < \gamma \text{ a } \pi - \gamma < \omega t \leq \pi \\ u(t) &= h & \gamma < \omega t \leq \pi - \gamma \end{aligned} \quad (2.33)$$

Výstup (rovnice 2.33) opět aproximujeme pomocí Fourierova rozvoje podle následujícího vztahu:

$$u(t) = \sum_{n=1}^{\infty} B_n \sin n \omega t \quad (2.34)$$

kde

$$B_n = \frac{2}{\pi} \int_0^{\pi} u(t) \sin n \omega t d(\omega t) \quad (2.35)$$

Velikost úhlu γ má vliv na přesnost získaných kritických parametrů. Mohou nastat následující případy:

- $0 < \gamma < \pi/2$ ($\infty > k > h/a$)

Pokud dosadíme rovnici 2.33 do rovnice 2.35 dostaneme:

$$B_n = \frac{2h}{\pi} \left[\frac{1}{\sin \gamma} \left(\frac{\sin(1-n)\gamma}{1-n} - \frac{\sin(1+n)\gamma}{1+n} \right) - \frac{1}{n} (\cos(\pi - \gamma) - \cos \gamma) \right], n = 1, 3, \dots \quad (2.36)$$

$, n = 2, 4, \dots$

Když uvažujeme pouze první harmonickou, dostaneme následující aproximaci výstupu z relé:

$$N(a) = \frac{2h}{\pi \bar{a}} \left[\left(\sin^{-1} \frac{\bar{a}}{a} \right) + \left(\frac{\bar{a}}{a} \sqrt{1 - \left(\frac{\bar{a}}{a} \right)^2} \right) \right] \quad (2.37)$$

kde $\bar{a} = h/k$

$N(a)$ je aproximace relé první harmonickou Fourierova rozvoje

- $\gamma = 0$ ($k \rightarrow \infty$)

V tomto krajním případě se saturační relé stává ideálním. Pokud spočítáme koeficient B_n z rovnice 2.36 pro první harmonickou, dostaneme vztah pro náhradu ideálního relé (rovnice 2.29):

$$N(a) = \frac{4h}{\pi a} \quad (2.38)$$

- $\gamma = \pi/2$ ($k = h/a$)

Nyní uvažujeme druhý krajní případ. Sklon saturačního relé je volen tak, že $\bar{a} = a$. Po dosazení $\gamma = \pi/2$ do rovnice 2.36 dostaneme:

$$\begin{aligned} B_n &= h & n &= 1 \\ B_n &= 0 & \text{jinak} \end{aligned} \quad (2.39)$$

O aproximaci se v tomto případě nedá mluvit, protože náhrada první harmonickou vyjadřuje přesně výstup z relé.

$$N(a) = \frac{h}{a} \quad (2.40)$$

V prvních dvou případech, pokud aproximujeme výstup z relé první harmonickou fourierova rozvoje, je náhrada přibližná. Je však tím přesnější, čím více se úhel γ blíží hodnotě $\pi/2$. Pokud se úhel rovná přesně hodnotě $\pi/2$, aproximace pomocí první harmonické je přesným vyjádřením výstupu.

2.3.2 Možný problém při použití saturačního relé

Pokud zvolíme špatný sklon saturačního relé, může nastat situace, že nedojde k rozkmitání řízené soustavy kritickou periodou kmitů. Podmínka pro vznik oscilací je následující:

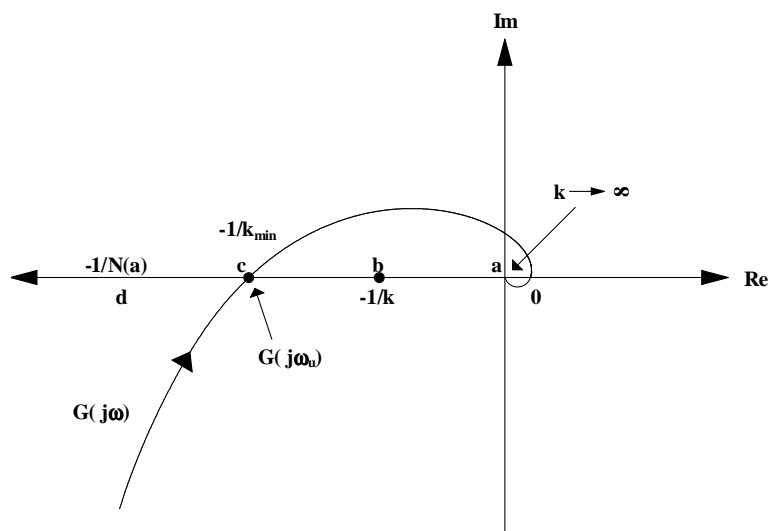
$$1 + G(j\omega_u)N(a) = 0 \Rightarrow G(j\omega_u) = -\frac{1}{N(a)} \quad (2.41)$$

kde $N(a)$ je aproximace relé pomocí první harmonické Fourierova rozvoje,

$G(j\omega_u)$ je bod frekvenční charakteristiky řízené soustavy v komplexní rovině při kritické frekvenci.

Rovnici 2.41 můžeme vyřešit graficky v komplexní rovině jako průsečík dvou křivek, který odpovídá kritickému bodu (K_u a ω_u). Pro ideální relé začíná charakteristika $-1/N(a)$ v bodě 0 (bod a na Obr. 2.9) a jde k $-\infty$. Pokud použijeme saturační relé, počáteční bod charakteristiky $-1/N(a)$ odpovídá hodnotě $-1/k$. Se snižujícím se sklonem k saturačního relé se počáteční bod posunuje stále více doleva po záporné reálné ose. Pokud výraz $-1/k$ odpovídá bodu b , k meznímu cyklu ještě dojde. Když však snížíme sklon k natolik, že počáteční bod charakteristiky $-1/N(a)$ je vlevo od bodu c , k meznímu cyklu nedojde. Existuje proto minimální sklon k_{\min} , při

kterém ještě dojde k meznímu cyklu. Sklon relé proto musíme volit tak, aby došlo k meznímu cyklu a zároveň se odhadované kritické parametry dostatečně zpřesnily.



Obr. 2.9 Grafické řešení rovnice 2.41

3. ŘÍDÍCÍ ALGORITMY

3.1 DISKRÉTNÍ PID REGULÁTOR S FILTRACÍ DERIVAČNÍ SLOŽKY A ANTIWINDUPEM

S převodem ze spojitě do diskrétní oblasti není problém u proporcionální a integrační části PID regulátoru. Integrační složka se nahradí sumací pomocí obdélníků zleva.

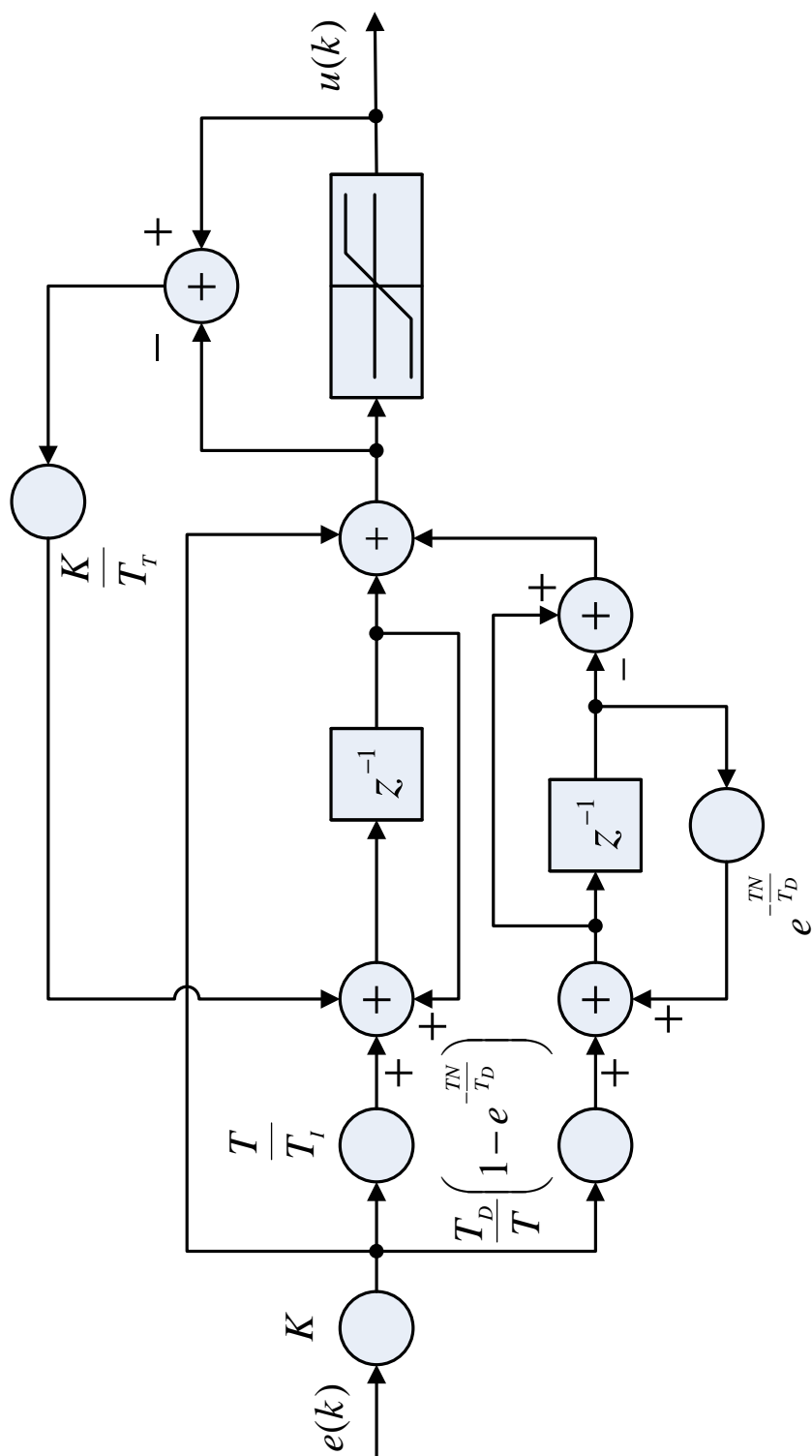
Derivační složku však nestačí pouze nahradit diferencí. Je nutné ji filtrovat, aby nezesilovala vysoké frekvence, tzn. šum. Je použita filtrace navržená podle [7]. Přenosovou funkci derivační složky můžeme získat z následující rovnice.

$$F_D(z) = (1 - z^{-1})Z\{h(k)\} \quad (3.1)$$

Z rovnice vidíme, že si můžeme určit požadavky na odezvu na jednotkový skok ($h(k)$) derivační části regulátoru. Tyto požadavky vycházejí z vlastností odezvy spojitě filtrované derivační složky na jednotkový skok. Prvním požadavkem je, aby odezva měla maximum v kroku $k = 0$. Poté by měla exponenciálně klesat. Druhým požadavkem je, aby plocha vymezená odezvou na jednotkový skok byla rovna právě derivační konstantě regulátoru T_D . Pokud se tyto požadavky vyjádří matematicky a spojí dohromady, vznikne přenosová funkce pro derivační část regulátoru.

$$F_D(z) = \frac{T_D}{T} (1 - e^{-\frac{NT}{T_D}}) \frac{1 - z^{-1}}{1 - e^{-\frac{NT}{T_D}} z^{-1}} \quad (3.2)$$

Do regulátoru je ještě zahrnut dynamický antiwindup, který zajišťuje omezení přebuzení integrační složky. Funkce je zřejmá z Obr. 3.1. Pokud je akční orgán schopen zpracovat akční zásah z regulátoru, je difference mezi akčním zásahem z regulátoru a akčním zásahem vystupujícím z akčního členu nulová. Proto se na vstup sumace nic nepřičítá. Pokud se však akční orgán dostane do saturace, je záporně vzatá difference přes konstantu K/T_T přičítána na vstup sumace. Rovnice 3.3 vyjadřuje přenosovou funkci popsaného regulátoru.



Obr. 3.1: Diskrétní PID regulátor s filtrací derivační složky a antiwindupem

$$F_R(z) = K \left(1 + \frac{Tz^{-1}}{T_I(1-z^{-1})} + \frac{T_D}{T} (1 - e^{-\frac{NT}{T_D}}) \frac{1-z^{-1}}{1 - e^{-\frac{NT}{T_D}} z^{-1}} \right) \quad (3.3)$$

kde K je proporcionální zesílení

T je perioda vzorkování

T_I je integrační konstanta regulátoru

T_D je derivační konstanta regulátoru

N konstanta omezující zesílení rušivých signálů na vyšších frekvencích (volí se v rozmezí 3 – 20)

3.2 OMEZENÍ PŘEKMITU REGULOVANÉ VELIČINY

Pokud je požadavek na aperiodický přechodový děj, máme v podstatě tři možnosti [8]. Změnu struktury PID regulátoru na I-PD nebo PI-D regulátor, Feed-Forward nebo filtraci žádané hodnoty.

I-PD regulátor se od diskrétního PID regulátoru liší tím, jaká veličina vstupuje do jednotlivých větví regulátoru. Do sumační větve vstupuje i nadále regulační odchylka, kvůli nulové ustálené hodnotě. Do derivační větve vstupuje záporně vzatá regulovaná veličina. Do proporcionální větve vstupuje také záporně vzatá regulovaná veličina, ke které se přičítá žádaná hodnota, násobená koeficientem β . Tento koeficient může nabývat hodnot z intervalu $\langle 0,1 \rangle$. Akční zásah se počítá podle následující rovnice:

$$U(z) = K(\beta W(z) - Y(z) + \frac{Tz^{-1}}{T_I(1-z^{-1})} E(z) - \frac{T_D}{T} (1 - e^{-\frac{NT}{T_D}}) \frac{1-z^{-1}}{1 - e^{-\frac{NT}{T_D}} z^{-1}} Y(z)) \quad (3.4)$$

kde β je koeficient, který určuje, jak moc se do proporcionální větve přičítá žádaná hodnota.

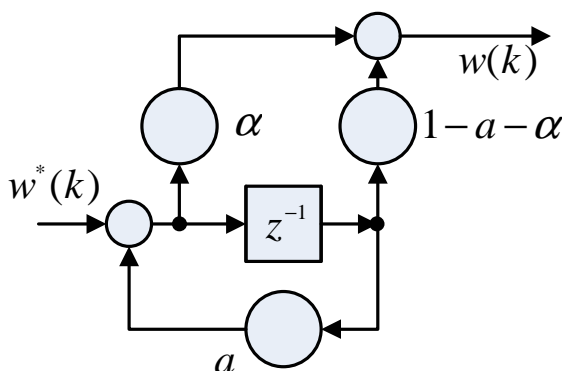
Struktura regulátoru je podobná jako struktura regulátoru na Obr. 3.1, liší se pouze vstupy do jednotlivých větví.

Feed-Forward, neboli dopředná vazba, má význam hlavně v zařízeních, kde se nemůže měnit struktura regulátoru a je potřeba omezit překmit regulované veličiny. Proto se přidá vazba od žádané hodnoty k vypočítanému akčnímu zásahu.

Při ověřování správnosti naprogramování I-PD regulátoru a Feed-Forward regulátoru v simulinku bylo zjištěno, že jsou oba algoritmy v podstatě stejné. Aby byly odezvy totožné, musí být v jednom regulátoru nastavena hodnota parametru β na β_1 a v druhém na hodnotu $1 - \beta_1$. Implementován byl pouze I-PD regulátor.

Třetí možností, jak omezit překmit regulované veličiny, je filtr žádané hodnoty. Jeho struktura je na Obr. 3.2, kde a je substituce za $\exp(-T/T_1)$. Diskrétní přenos je dán vztahem:

$$F_w(z) = \frac{\alpha + (1 - e^{-\frac{T}{T_1}} - \alpha)z^{-1}}{1 - e^{-\frac{T}{T_1}}z^{-1}} \quad (3.5)$$



Obr. 3.2 Filtr žádané hodnoty

Koeficient α určuje, jak moc bude žádaná hodnota filtrována a nastavuje se z intervalu $\langle 0,1 \rangle$. Časová konstanta filtru T_1 se často volí jako $T_D + T_I$.

3.3 BEZNÁRAZOVÉ PŘEPÍNÁNÍ

U průmyslových regulátorů, které jsou v dnešní době realizovány pouze v číslicové formě, je k dispozici více řídicích algoritmů. Samozřejmostí je možnost ovládat proces ručně. Z tohoto důvodu je nutné beznárazové přepnutí mezi řídicími algoritmy a ručním řízením. Musí být splněn následující požadavek. Při přechodu z jednoho řídicího algoritmu na druhý, má být změna akčního zásahu menší, než je technologicky přijatelná hodnota.

Přepnutí se děje zásadně při nulové regulační odchylce. Při nenulové regulační odchylce není možné beznárazového přepnutí dosáhnout. Každý regulátor

má totiž jinou hodnotu akčního zásahu pro stejnou hodnotu odchylky. Beznárazové přepnutí mezi dvěma řídicími algoritmy má pro každou dvojici svá specifika a musí být řešeno individuálně.

4. STRUKTUROVANÝ NÁVRH SOFTWARE

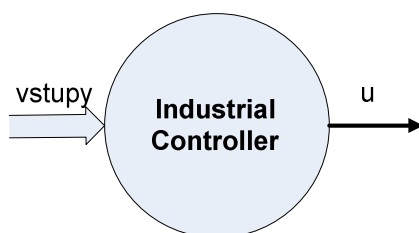
Typickými kroky při vytváření softwaru jsou následující [9]:

- analýza požadavků
- specifikace softwaru
- návrh architektury softwaru
- implementace
- testování
- dokumentace
- údržba

První dva kroky nebyly součástí práce, protože již plynou ze zadání. Třetí krok (návrh architektury softwaru) je hlavní částí, která rozhoduje o úspěchu celého projektu. Pokud se této fázi věnuje dostatečné množství času, následná implementace je snadnější a trvá kratší dobu. Údržba i přehlednost programu také značně závisí na dobrém softwarovém návrhu.

Jedním z přístupů, který se používá pro návrh softwaru je metoda strukturovaného návrhu softwaru. Podstata spočívá v tom, že problém je postupně dělen na menší a menší podproblémy. Mezi těmito podproblémy se stanovují vazby (rozhraní), pomocí nichž si jednotlivé podproblémy vyměňují data a informace důležité pro chod programu. Data by měla být v programu organizována ve strukturách. Dělení končí tehdy, kdy je jednodušší dílčí podproblém naprogramovat, než ho znovu rozdělit. Dílčí problémy se v automatizačních úlohách většinou implementují jako konečné stavové automaty.

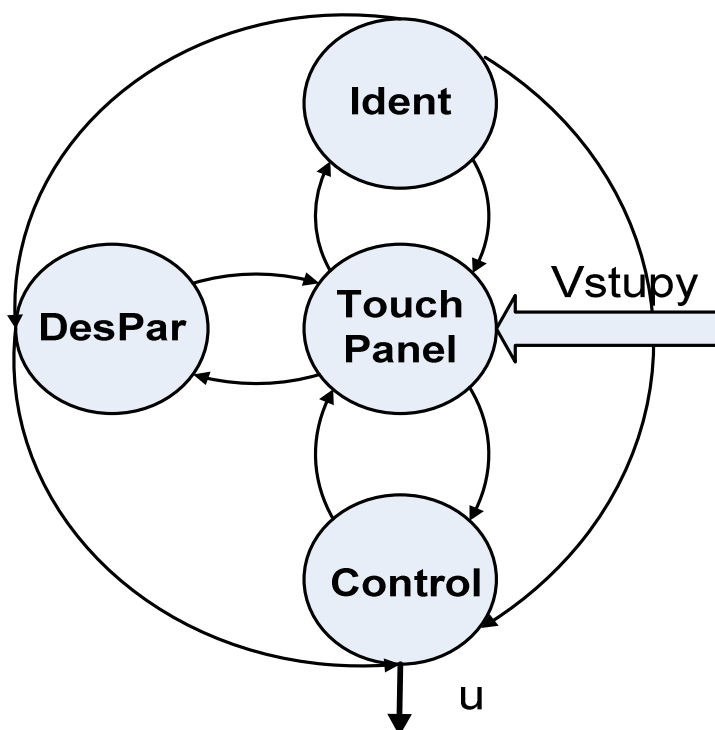
Nejvyšší vrstva regulačního členu je na Obr. 4.1.



Obr. 4.1 Architektura softwaru v nejvyšší vrstvě

Jako vstupy do regulátoru jsou uvažovány: žádaná hodnota, měřená regulovaná veličina, řád identifikovaného modelu a celá řada dalších. Výstupem z regulátoru je akční zásah. Tento akční zásah působí prostřednictvím akčního členu na regulovanou soustavu.

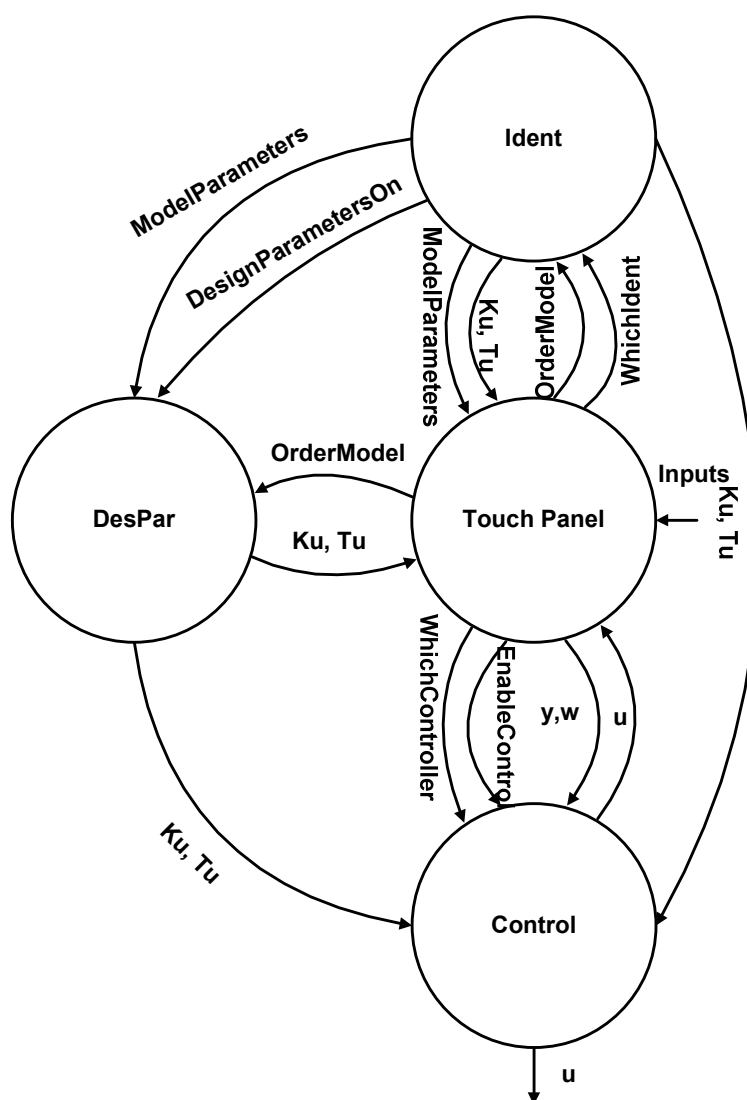
Ve druhé vrstvě (Obr. 4.2) je regulátor rozdělen do základních funkčních částí, které plynou více méně ze zadání a jsou ustanoveny vazby mezi jednotlivými bloky.



Obr. 4.2 Druhá vrstva návrhu

Jak je vidět, problém se rozpadl na čtyři dílčí podproblémy, které je už jednodušší obsáhnout.

V další vrstvě (Obr. 4.3) je přesně specifikováno rozhraní mezi jednotlivými bloky z druhé vrstvy. Z obrázku je zřejmé, pomocí jakých proměnných spolu jednotlivé bloky komunikují. Hlavním členem v celé struktuře je dotyková obrazovka (Touch Panel), která zajišťuje komunikaci mezi uživatelem a ostatními bloky regulátoru. Uživatel zadává všechny žádané vstupy, které jsou důležité pro chod regulátoru.



Obr. 4.3 Třetí vrstva návrhu

Bloku Ident předává uživatel informaci jakou metodou se má identifikovat proces. V případě identifikace metodou nejmenších čtverců je třeba ještě zadat řád modelu, který se má identifikovat. Řád modelu se předává také bloku DesPar, aby tento blok věděl, podle jakého algoritmu má počítat kritické parametry. Blok Ident předává zpět na obrazovku parametry identifikovaného modelu. Tyto parametry jsou také vstupem do bloku DesPar pro výpočet kritické periody kmitů a kritického zesílení. V případě identifikace pomocí relé ve zpětné vazbě jsou kritické parametry rovnou počítány v bloku Ident. Proto jsou posílány na obrazovku jako informace pro uživatele a zároveň také do bloku Control pro výpočet konstant regulátoru.

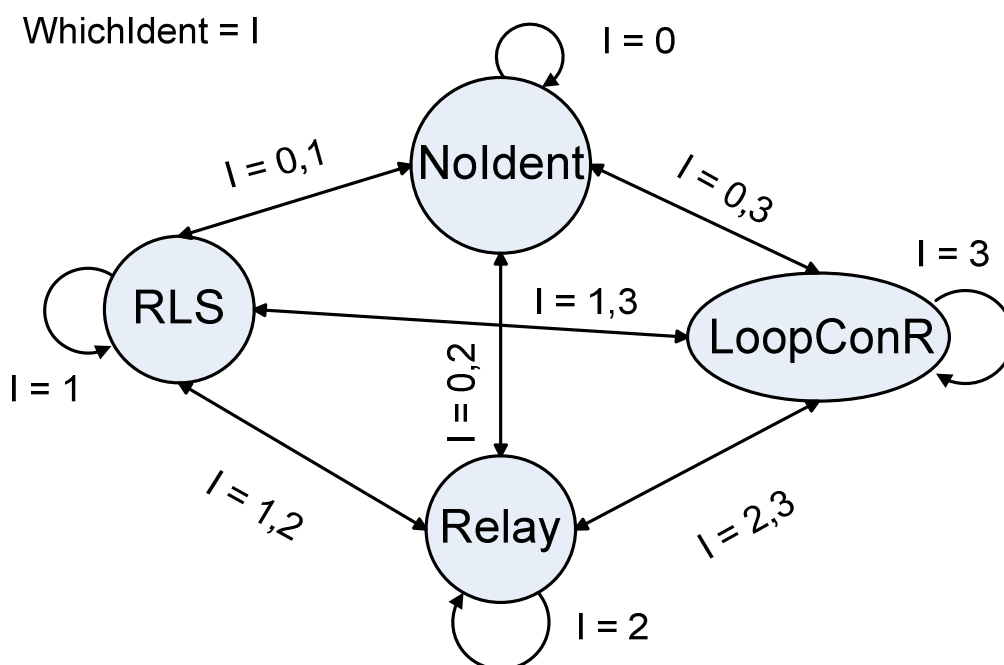
Do bloku Control předává uživatel informaci o žádané hodnotě. Aktuální hodnota regulované veličiny je předávána bloku Control přímo z řízeného procesu. Dále uživatel zadává, zda se má vůbec řídit, tzn. jestli je řízení některým z dostupných algoritmů dovoleno. Posledním předávaným parametrem je, jaký algoritmus se pro řízení použije, nebo zda bude řízení provádět ručně sám uživatel. Blok Control předává zpět na obrazovku hodnotu akčního zásahu pro zobrazení.

Blok DesPar potřebuje ke své činnosti (výpočet kritických parametrů modelu) vědět, jestli už jsou tyto parametry známy. O tom mu dává zprávu blok Ident pomocí proměnné DesignParametersOn. Poté je kritické zesílení a kritická perioda kmitů spočítány a poslány do bloku Control pro další zpracování.

Dělení do dalších vrstev nemá smysl, protože se již dá pro každý navržený blok nakreslit stavový nebo vývojový diagram a naimplementovat jej.

4.1 BLOK IDENT

Blok identifikace lze popsat pomocí stavového automatu na Obr. 4.4.



Obr. 4.4 Stavový diagram pro blok Ident

Jak je vidět, blok může být ve čtyřech stavech: NoIdent – bez identifikace, RLS – identifikace metodou nejmenších čtverců, Relay – identifikace pomocí relé ve zpětné vazbě a LoopConR – identifikace a řízení pomocí knihovních funkcí LoopConR od firmy B&R. Po spuštění regulačního členu je jako výchozí stav zvolen NoIdent, tzn. identifikace se neprovádí. Mezi jednotlivými stavy se dá plynule přecházet prostřednictvím proměnné WhichIdent. Pokud se identifikuje pomocí relé ve zpětné vazbě, tak po dokončení identifikace blok automaticky přechází do stavu NoIdent.

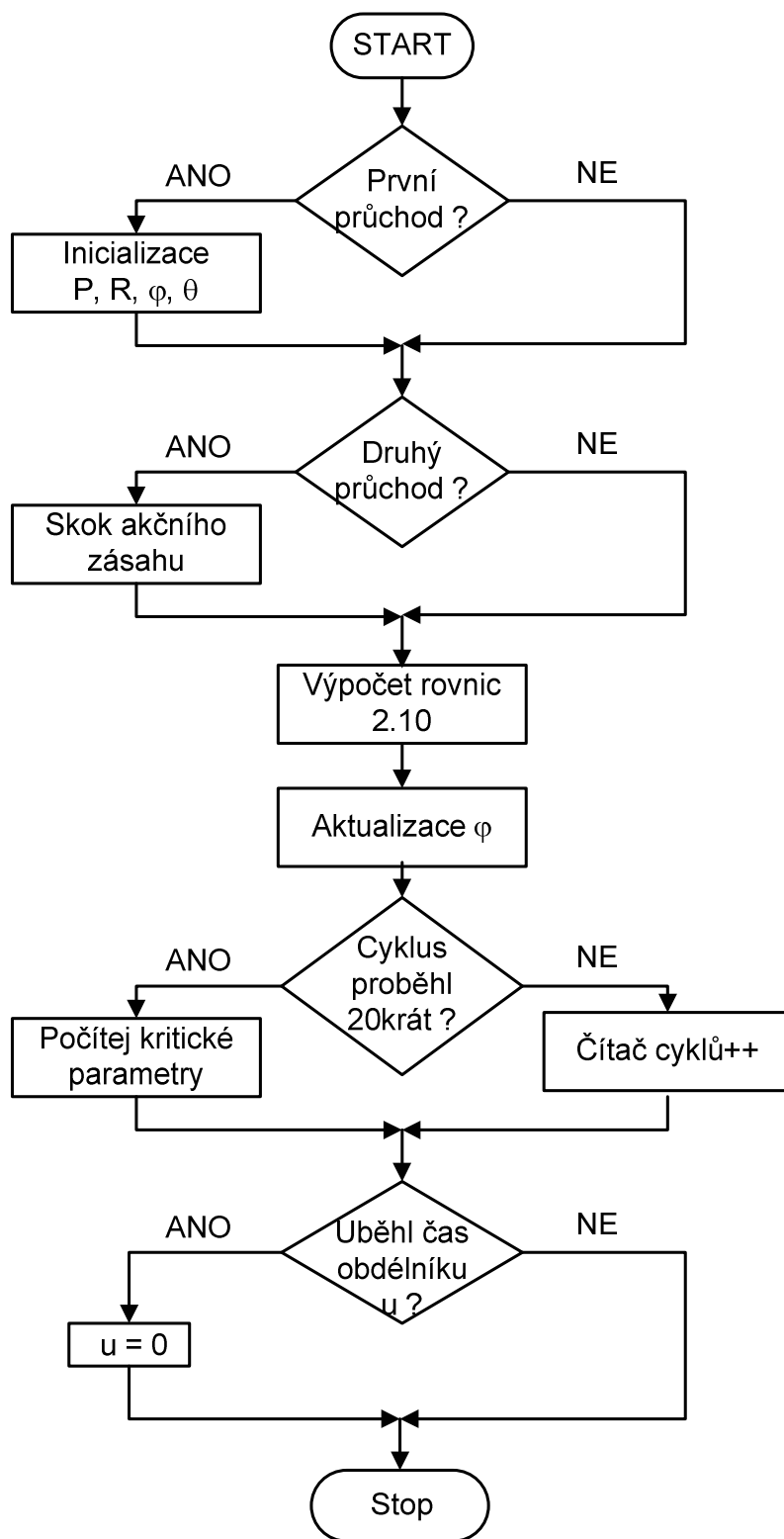
4.1.1 Rekurzivní metoda nejmenších čtverců a návrh parametrů

Rekurzivní metoda nejmenších čtverců je implementována se směrovým zapomínáním podle kapitoly 2.1.1. Jeden krok algoritmu je na Obr. 4.5. V prvním kroku se provádí inicializace matic R , P a vektorů φ , θ . Jako budící signál se používá jednotkový skok, ke kterému dojde v druhém průchodu. Jeho amplitudu si může uživatel zvolit. Po skoku akčního zásahu se již provádí výpočet podle rovnic 2.10. Dále se v každém cyklu aktualizuje vektor φ . Po dvaceti sekundách akční zásah přejde zpět na nulu. Po dvaceti průchodech se nastaví proměnná DesignParametersOn, která dává signál bloku DesPar, že má podle nalezených parametrů modelu počítat kritické parametry řízené soustavy. Přechodem na jiný typ identifikace (změna stavu na Obr. 4.4) a opětovném navrácení do metody nejmenších čtverců, se provádí identifikace opět od prvního průchodu.

Blok DesPar počítá kritické parametry řízené soustavy. Pro první řád podle rovnic 2.21 a 2.23, pro druhý řád se počítá podle vývojového diagramu z Obr. 2.3 a pro třetí řád dle Obr. 2.4. Proměnná OrderModel určuje řád identifikovaného modelu, proto hodnota proměnné rozhoduje, podle jakého postupu se budou kritické parametry počítat.

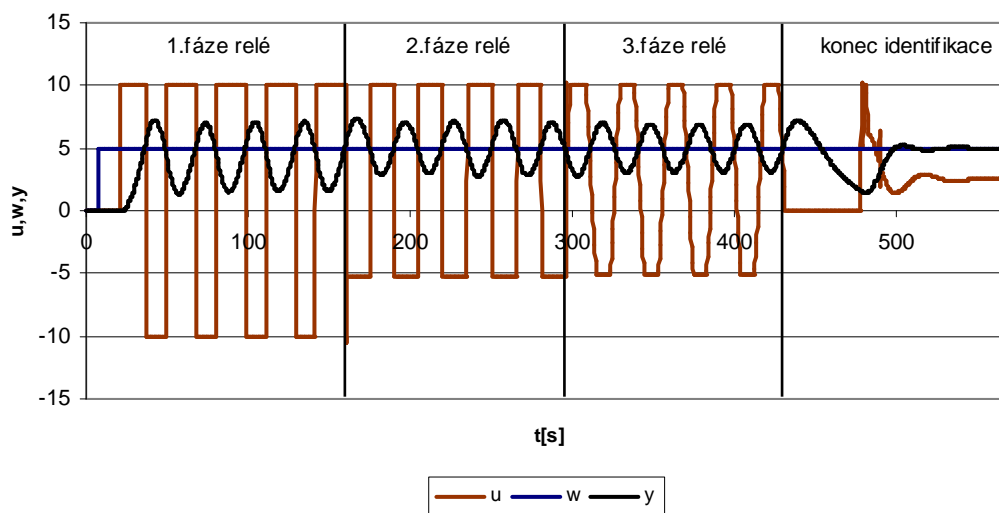
4.1.2 Identifikace pomocí relé ve zpětné vazbě

Identifikace pomocí relé ve zpětné vazbě probíhá ve třech fázích (Obr. 4.6). Každá fáze trvá čtyři periody mezního cyklu regulované soustavy. V první fázi se řízená soustava rozkmitá pomocí ideálního relé, které přechází mezi stavy UMAX a UMIN, což jsou hodnoty minimálního a maximálního akčního zásahu, podle typu



Obr. 4.5 Vývojový diagram pro identifikaci pomocí RLS

Průběhy veličin při identifikaci pomocí relé ve zpětné vazbě



Obr. 4.6 Průběh identifikace pomocí relé ve zpětné vazbě

akčního členu. V první fázi se měří statické zesílení identifikované soustavy. To je později použito pro optimalizaci maximálního a minimálního výstupu z relé v druhé fázi tak, aby střední hodnota výstupní regulované veličiny měla hodnotu žádané hodnoty w .

Statické zesílení K_s je poměr střední hodnoty regulované veličiny a střední hodnoty akčního zásahu z relé. Střední hodnota regulované veličiny je počítána jako průměr maximální a minimální hodnoty. Střední hodnota akčního zásahu je počítána podle vztahu:

$$u_{stř.} = \frac{1}{T} \int_0^T u(t) dt \quad (4.1)$$

Ve druhé fázi je rozkmit výstupu z relé optimalizován tak, že střední hodnota je rovna [10]:

$$u_{stř.} = \frac{w}{K_s} \quad (4.2)$$

a rozkmit je maximální, jaký dovolí akční člen. Pokud totiž bude střední hodnota výstupu z relé rovna hodnotě podle rovnice 4.2, bude výstupní regulovaná veličina

kmitat symetricky kolem žádané hodnoty, jak je vidět z Obr. 4.6. V této fázi se již spočítá první odhad kritického zesílení identifikované soustavy podle vztahu 2.29.

Ve třetí fázi se použije saturační relé, jak bylo vysvětleno v kapitole 2.3.1. Byl použit postup navržený v [6] tak, aby k meznímu cyklu ještě došlo a zároveň byly zpřesněny parametry. Pro saturační relé se zvolí sklon lineární části jako:

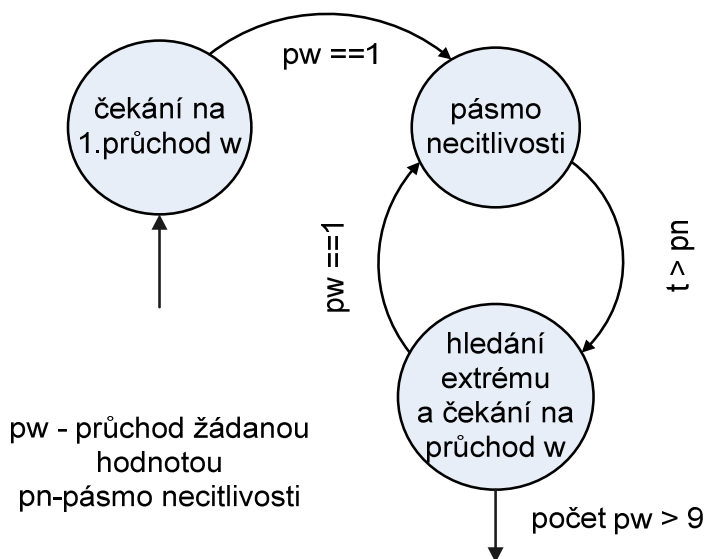
$$k = 1,4K_u \quad (4.3)$$

kde K_u představuje odhad kritického zesílení, získaný v druhé fázi identifikace pomocí relé ve zpětné vazbě

Z Obr. 4.6 je vidět, že je ve třetí fázi použito saturační relé. Po dokončení čtyř period je změřena kritická perioda kmitů a spočítáno kritické zesílení podle vztahu:

$$K_u = \frac{2h}{\pi \bar{a}} \left[\left(\sin^{-1} \frac{\bar{a}}{a} \right) + \left(\frac{\bar{a}}{a} \sqrt{1 - \left(\frac{\bar{a}}{a} \right)^2} \right) \right] \quad (4.4)$$

V každé fázi je potřeba měřit amplitudu výstupní regulované veličiny a periodu kmitů. Perioda kmitů se měří pomocí zaznamenávání průchodů žádanou hodnotou (stejný princip jako při průchodu nulou). Pro účely určení amplitudy regulované veličiny se zaznamenává extrém mezi dvěma průchody nulou. Celé měření probíhá podle stavového automatu na Obr. 4.7.



Obr. 4.7 Stavový automat pro měření kritických parametrů

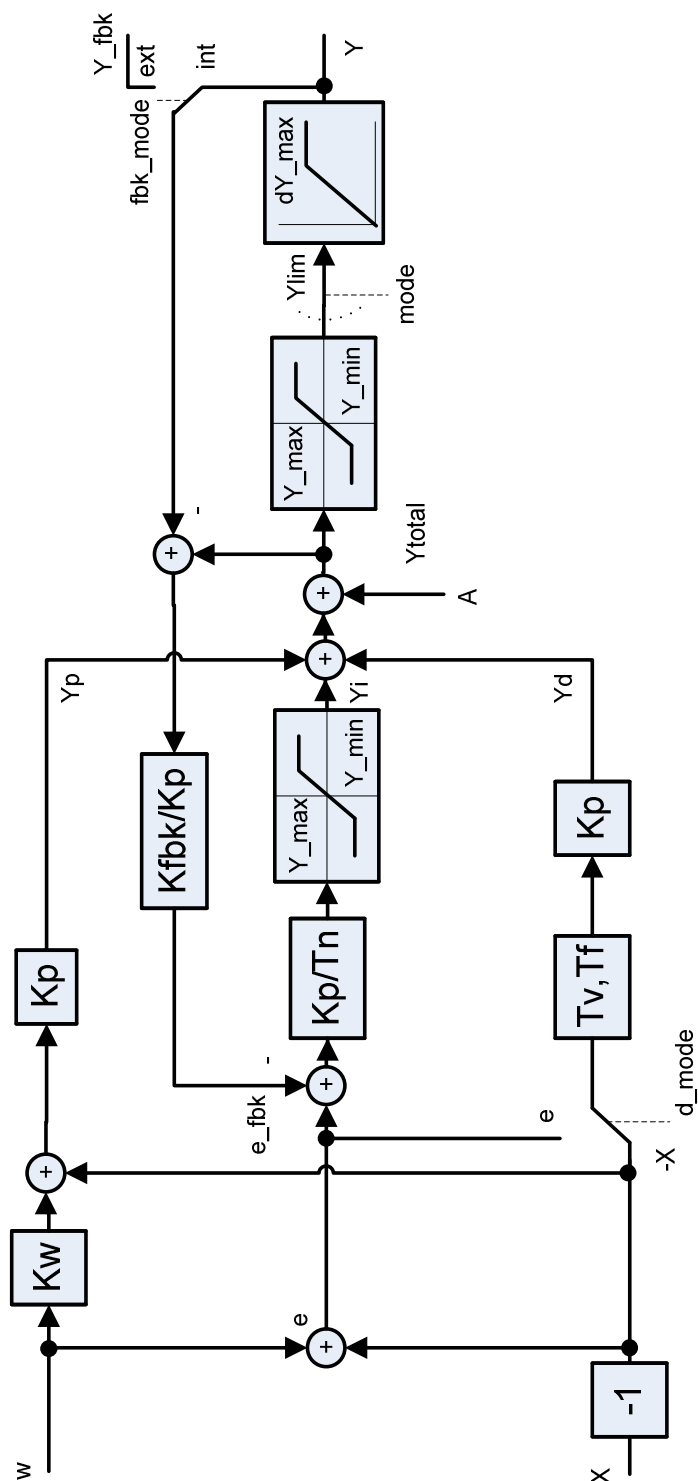
Nejprve se čeká na první průchod žádanou hodnotou. Po každém průchodu žádanou hodnotou, kterých je pro čtyři periody devět, se přejde do stavu pásma necitlivosti. Tento stav slouží k následujícímu. Pokud regulovaná veličina obsahuje šum, může překmitnout žádanou hodnotu v několika po sobě následujících periodách. Pokud je však měření ve stavu pásma necitlivosti, průchod žádanou hodnotou se nedetekuje. Šířku pásma necitlivosti může uživatel nastavit. Po uplynutí času delšího než je pásmo necitlivosti měření, přechází do stavu hledání extrému (maxima nebo minima) a detekce dalšího průchodu žádanou hodnotou. Pokud se detekuje devět průchodů žádanou hodnotou, identifikace přechází do další fáze nebo končí.

4.1.3 Identifikace a řízení pomocí knihovny LoopConR

Blok byl implementován kvůli porovnání standardního adaptivního regulátoru firmy B&R s navrženým regulačním členem. Adaptivní regulátor je dostupný v knihovně LoopConR, která je součástí Automation Studia, jako funkční blok (dále jen FB) LCRSlimPID() [10]. Tento FB používá FB LCRPID() jako regulátor a FB LCRPIDTune() pro identifikaci. Vstupními parametry do LCRSlimPID() jsou *enable* (vykonávání/nevykonávání FB), *w* (žádaná hodnota), *x* (regulovaná veličina), *request* (specifikace identifikační metody/ruční nastavení parametrů regulátoru) a *pPar* (parametry pro regulátor). Regulátor začne pracovat poté, co je dokončena identifikace nebo jsou nastaveny konstanty regulátoru.

FB LCRPID() implementuje PID regulátor, jehož blokové schéma je na Obr. 4.8. Význam jednotlivých konstant, signálů a módů je shrnut v Tab. 4.1. PID regulátor může pracovat v ručním nebo automatickém režimu. Jako vstupní signál do derivační složky regulátoru může být zvolena regulační odchylka nebo záporná regulovaná veličina (analogie s I-PD regulátorem). Signál pro omezení přebuzení integrační složky je možno odvodit od výstupu z akčního členu nebo z pevně stanovené hodnoty. Model akčního členu lze určit nejen maximálním a minimálním akčním zásahem, ale i maximálním nárůstem akčního zásahu.

K identifikaci je použit FB LCRPIDTune(), který umožňuje identifikaci pomocí dvou metod. První je identifikace pomocí relé ve zpětné vazbě, která může



Obr. 4.8 Blokové schéma PID regulátoru firmy B&R

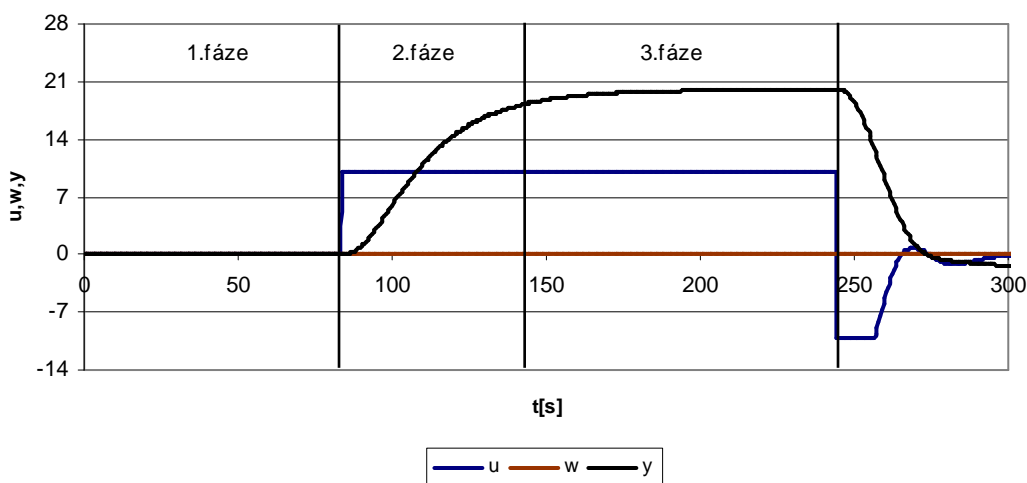
w	žádaná hodnota	Kw	zesílení žádané hodnoty
e	regulační odchylka	Kp	zesílení regulátoru
x	regulovaná veličina	kfbk	zesílení signálu pro antiwindup
e_fbk	signál pro antiwindup po zesílení	Tn	integrační konstanta
Yp	proporcionální akční zásah	Tv	derivační konstanta
Yi	integrační akční zásah	Tf	filtrační konstanta derivační složky
Yd	derivační akční zásah	Y_max	maximální akční zásah
Ytotal	celkový akční zásah	Y_min	minimální akční zásah
A	přímá hodnota akčního zásahu	dY_max	maximální přírůstek akčního zásahu
Ylim	akční zásah za akčním členem	d_mode	mód derivační složky
Y	akční zásah do soustavy	mode	mód regulátoru
Y_fbk	signál pro antiwindup	fbk_mode	mód antiwindupu

Tab. 4.1 Význam symbolů PID regulátoru firmy B&R

probíhat až ve třech fázích, stejně jako v kapitole 4.1.2. Má ale spoustu dalších volitelných parametrů, které je možno najít v nápovědě Automation Studia.

Druhou možností, jak identifikovat pomocí FB LCRPIDTune(), je odezva systému na jednotkový skok. Identifikace může mít až tři fáze (Obr. 4.9), podle toho jestli se navrhují parametry na sledování žádané hodnoty nebo na odstranění poruchy.

Průběhy veličin při identifikaci pomocí skoku pomocí FB LCRSlmPID z knihovny LoopConR

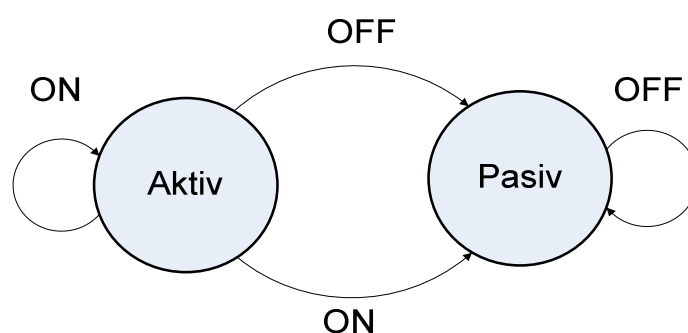


Obr. 4.9 Fáze identifikace pomocí odezvy na jednotkový skok

V první fázi se detekuje ustálený stav regulované veličiny na hodnotě žádané hodnoty. Akční zásah proto musí být $u = w/K_s$, kde K_s je statické zesílení soustavy. Ustálený stav je detekován tak, že regulovaná veličina musí mít po určitou dobu rozkmit okolo žádané hodnoty menší, než je nastavené pásmo. V druhé fázi dojde ke skoku akčního zásahu. Z průběhu regulované veličiny se určuje maximální nárůst regulované veličiny a zpoždění. Pokud je požadováno regulátor nastavit na odezvu na poruchu, je po této fázi identifikace ukončena. Ve třetí fázi se měří čas, než se systém znovu ustálí. Tato doba je potom použita pro výpočet parametrů regulátoru pro sledování žádané hodnoty. I při této metodě je možné nastavit spoustu parametrů tak, aby identifikace vyhovovala dané regulované soustavě.

4.2 BLOK CONTROL

Blok Control obsahuje tři možnosti řízení. Manuální řízení, algoritmus PID regulátoru s omezením derivační složky a antiwindupem a I-PD regulátor. Blok je možné povolit nebo zakázat pomocí globální proměnné EnableControl. Akční zásah PID regulátoru se počítá podle rovnice 3.3 a akční zásah I-PD regulátoru podle 3.4. Manuální řízení se provádí pomocí tlačítek více méně z dotykové obrazovky. Každé z možných typů řízení může být v aktivním nebo pasivním režimu (Obr. 4.10).



Obr. 4.10 Stavový automat pro blok Control

Aktivní režim znamená, že řídicí algoritmus nebo ruční řízení přímo produkuje akční zásah. Naproti tomu pasivní režim značí, že algoritmus sleduje akční zásah od aktivního regulátoru, aby byl připraven na případné přepnutí do aktivního

stavu. Toto zajistí beznárazové přepnutí mezi řídicími algoritmy. Aktivní může být pouze jeden typ řízení, ostatní jsou v tu dobu v pasivním stavu.

Parametry obou regulátorů se počítají z kritického zesílení a kritické periody kmitů regulované soustavy podle tabulek od Zieglera-Nicholse [11]. Lze si vybrat mezi klasickým nastavením podle ZN, nebo nastavením pro omezení kmitavého průběhu. Je samozřejmě možné parametry regulátoru nastavit libovolně na základě zkušeností a požadavků uživatele.

4.2.1 Beznárazové přepínání

Pokud má dojít k beznárazovému přepnutí mezi regulačními algoritmy, musí tyto algoritmy ve svém pasivním režimu nastavovat svoje vnitřní stavy (sumaci, diferenci a manuální hodnotu akčního zásahu) tak, aby v okamžiku po přepnutí byl akční zásah stejný jako u aktivního regulátoru v okamžiku před přepnutím. Jak již bylo dříve zmíněno, k přepínání musí docházet zásadně při nulové regulační odchylce.

U ručního řízení je pasivní režim velmi jednoduchý. Pouze se do manuální hodnoty akčního zásahu ukládá aktuální hodnota akčního zásahu, vypočtená aktivním regulátorem. Pro zajištění beznárazového přepnutí z ručního řízení na jeden ze dvou regulátorů je potřeba ještě kopírovat do žádané hodnoty aktuální hodnotu regulované veličiny pro zajištění nulové regulační odchylky při přepnutí.

U PID regulátoru je situace také velmi jednoduchá. Do všech tří větví tohoto paralelního regulátoru vstupuje regulační odchylka. Stačí proto aktuální akční zásah ukládat do sumace PID regulátoru a diferenci PID regulátoru nulovat. Pokud je totiž regulační odchylka nulová, bude příspěvek od proporcionální i derivační větve v okamžiku přepnutí nulový a díky tomu se bude aktuální akční zásah rovnat minulému akčnímu zásahu.

U I-PD regulátoru je situace složitější, protože do proporcionální i derivační větve vstupuje záporně vzatá hodnota regulované veličiny. Akční zásah u I-PD regulátoru se počítá podle následující rovnice:

$$u = K(\beta w - y) + sum + dif(e^{-\frac{TN}{T_D}} - 1) - Ky \frac{T_D}{T} (1 - e^{-\frac{TN}{T_D}}) \quad (4.5)$$

Pokud si z této rovnice vyjádříme proměnnou sum (rovnice 4.5) a tento výraz budeme do proměnné sum v pasivním režimu ukládat, dojde k beznárazovému přepnutí, jestliže dále po přepnutí zůstanou proměnné sum a dif neměnné.

$$sum = u - K(\beta w - y) - dif \left(e^{\frac{TN}{T_D}} - 1 \right) + Ky \frac{T_D}{T} \left(1 - e^{\frac{TN}{T_D}} \right) \quad (4.6)$$

Proměnná sum se počítá podle následujícího vztahu:

$$sum(k) = sum(k-1) + Ke \frac{T}{T_I} \quad (4.7)$$

Z toho plyne, že pokud dojde k beznárazovému přepnutí při nulové regulační odchylce, proměnná sum zůstává stejná (nic se k ní nepřičítá). Proměnná dif se počítá podle rovnice:

$$dif(k) = -Ky \frac{T_D}{T} \left(1 - e^{\frac{TN}{T_D}} \right) + dif(k-1) e^{\frac{TN}{T_D}} \quad (4.8)$$

Pokud požadujeme, aby dif v k -tém kroku a v $k-1$ kroku byla stejná, bereme ji v rovnici 4.8 jako identickou proměnnou a vyjádříme si ji. Tento výraz potom musíme v pasivním režimu ukládat do proměnné dif . Po vyjádření a úpravě dostaneme vztah:

$$dif = -Ky \frac{T_D}{T} \quad (4.9)$$

Pokud budeme v pasivním režimu I-PD regulátoru do proměnné sum I-PD regulátoru ukládat výraz podle rovnice 4.6 a do proměnné dif I-PD regulátoru výraz podle rovnice 4.9, bude přepnutí na tento regulátor beznárazové.

5. PRAKTICKÉ ASPEKTY IMPLEMENTACE

Navržený regulační člen byl implementován pomocí Automation Studia 3.0.71. Implementační řešení bylo rozděleno na tzv. tasky podle Obr. 4.2. Bloky Ident, Control a DesPar byly naprogramovány v jazyce ANSI C, zatímco blok Touch Panel byl vytvořen ve Visual Components Editoru. V ANSI C byly naprogramovány i další tasky nutné pro připojení regulátoru k reálnému procesu. Celé řešení funguje v zařízení Power Panel.

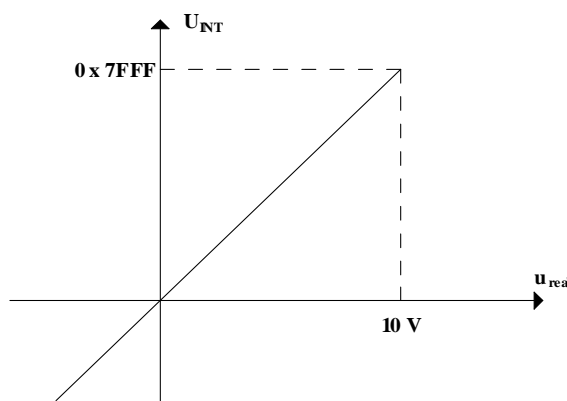
5.1 POUŽITÝ HARDWARE

Zařízení Power Panel 400 (PP400) integruje PLC spolu s dotykovou obrazovkou. Na tomto zařízení je nainstalován real-time operační systém Automation Runtime. PP400 vykonává nahrané tasky cyklicky s časovou smyčkou, která je určena taskovou třídou, ve které jsou tasky umístěny. Nejrychlejší tasková třída PP400 je 10 ms. Do této taskové třídy byly umístěny všechny tasky tvořící regulátor. Regulátor musí mít samozřejmě možnost změny periody vzorkování. Za běhu aplikace není možné umístění tasků v taskové třídě měnit, proto je použit přístup uvedený v [12]. Tasky jsou umístěny v nejrychlejší taskové třídě, proto je nejkratší možná perioda vzorkování rovna časové smyčce této taskové třídy (10 ms). Pro změnu periody vzorkování slouží dvě globální proměnné. Jednou z nich je proměnná, která čítá kolikrát byla vykonána 10 ms tasková třída od začátku. Pokud hodnota tohoto čítače násobená dobou taskové třídy je větší nebo rovna periodě vzorkování, je nastavena globální proměnná EnableTs, která povoluje vykonání všech tasků. Čítač je v tu chvíli nastaven na jedničku a čítání začíná znovu. Z toho vyplývá, že perioda vzorkování musí být násobkem časové smyčky taskové třídy (10 ms). K PP400 musejí být připojeny vstupně-výstupní karty, které zajistí spojení s reálnou soustavou.

5.1.1 Vstupně-výstupní karty

Hlavním výstupem z regulátoru je akční zásah, který řídí regulovanou soustavu. Převod akčního zásahu z digitální na analogovou veličinu zajišťuje karta X20AO4632. Tato karta má dvě možnosti výstupu ± 10 V a 0-20 mA a 16-bitový

číslicově-analogový převodník. Byl použit výstup ± 10 V pro ovládání vrtulky větráku. Do výstupní proměnné, která slouží k D/A převodu, nemůžeme zapsat přímo reálnou hodnotu akčního zásahu z programu, protože je typu INT. Akční zásah je proto třeba vyjádřit jako INT v příslušném rozsahu (Obr. 5.1).



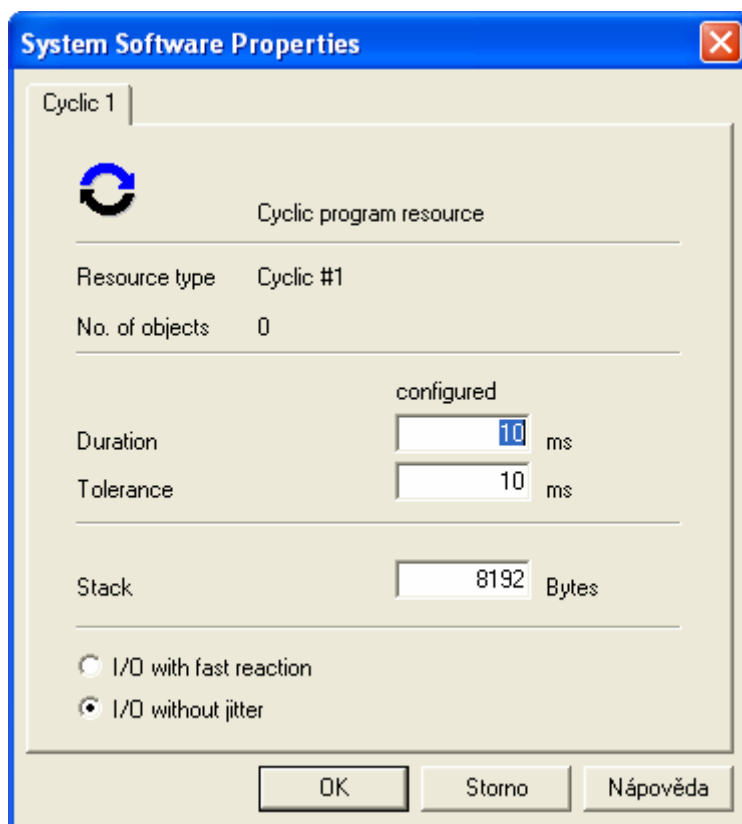
Obr. 5.1 Převod akčního zásahu jako reálné hodnoty na INT hodnotu

Maximální hodnota odpovídající deseti výstupním voltům je 0x7FFF (šestnáctkově). Mezi reálnou hodnotou a INT hodnotou platí přímá úměra, proto se převod děje podle následující rovnice.

$$u_{INT} = (INT) \left(\frac{0x7FFF}{10.0} \cdot u_{real} \right) \quad (5.1)$$

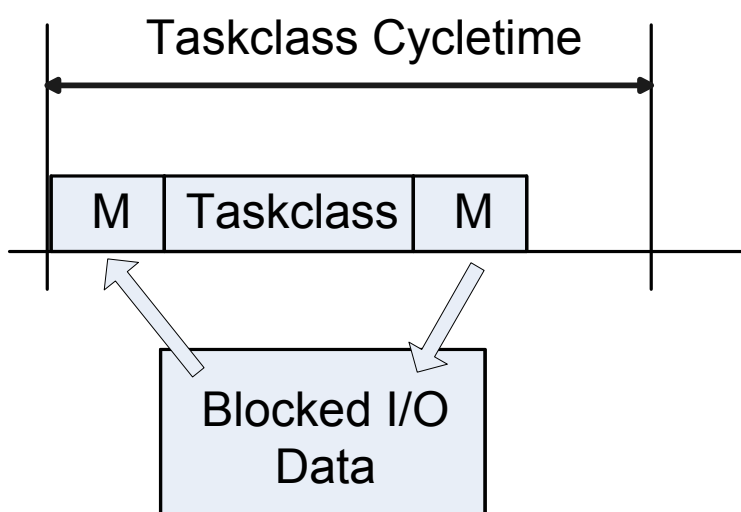
Tasková třídy 1 (Cyclic #1) má možnost nastavit jednu důležitou vlastnost [13] zápisu hodnot na výstup (Obr. 5.2). Můžeme si vybrat mezi dvěma volbami., I/O with fast reaction (Obr. 5.3) a I/O without jitter (Obr. 5.4). Pokud zvolíme první možnost, jsou výstupy zapsány hned poté, kdy je dokončen poslední task v taskové třídě. Druhá volba naopak zajistí zápis výstupů pokaždé ve stejný čas na konci doby vykonávání taskové třídy. Pokud má být perioda vzorkování deterministická, je nutné zvolit možnost I/O without jitter.

Informaci o velikosti regulované veličiny nám dává vstupní analogová karta X20AI2622, která umožňuje napěťový i proudový vstup. Pro vstup ± 10 V má rozlišení ± 12 bitů. Proměnná, do které se zapisuje velikost vstupního napětí, je typu INT, proto



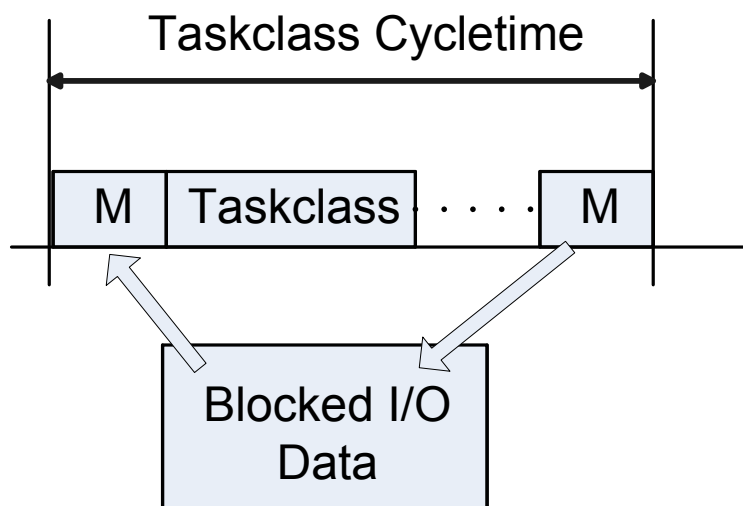
Obr. 5.2 Vlastnosti taskové třídy 1 (Cyclic #1)

M...I/O Mapping



Obr. 5.3 Volba I/O with fast response

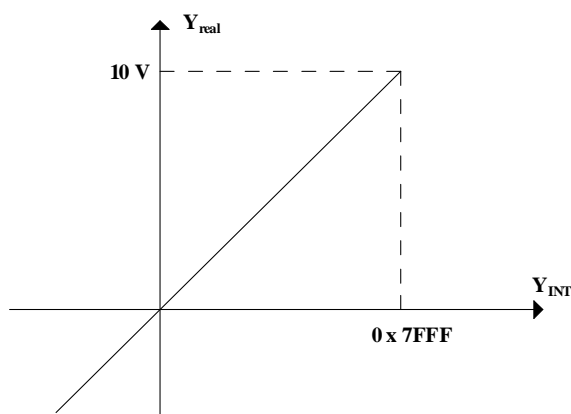
M...I/O Mapping



Obr. 5.4 Volba I/O without jitter

je nutný převod na reálnou hodnotu, použitelnou pro další zpracování v programu. Maximální hodnota z převodníku 0x7FFF odpovídá 10 V (Obr. 5.5). Přepoččet je podle následující rovnice:

$$y_{real} = (REAL)\left(\frac{10.0}{0x7FFF} \cdot y_{INT}\right) \quad (5.2)$$



Obr. 5.5 Převod regulované veličiny v INT reprezentaci do reálné reprezentace

Pokud by byla regulovaná veličina hodně zašuměná, je možné ji filtrovat. To je realizováno FB LCRMovAvgFlt(), který představuje filtr typu plovoucí průměr a

je z knihovny LoopConR. Hodnota regulované veličiny je počítána jako průměr z deseti hodnot. Dále je možné filtrovat žádanou hodnotu. Filtr žádané hodnoty je naprogramován podle Obr. 3.2. Filtr má jeden vnitřní stav, který musí být nastavován podle rovnice 5.3, pokud není filtr použit. Pokud by tomu tak nebylo a došlo by k zapnutí filtru, při nenulové nefiltrované žádané hodnotě, by došlo ke skoku filtrované žádané hodnoty.

$$stav = \frac{w^*}{1 - e^{-\frac{T}{T_l}}} \quad (5.3)$$

6. VIZUALIZACE A OVLÁDÁNÍ

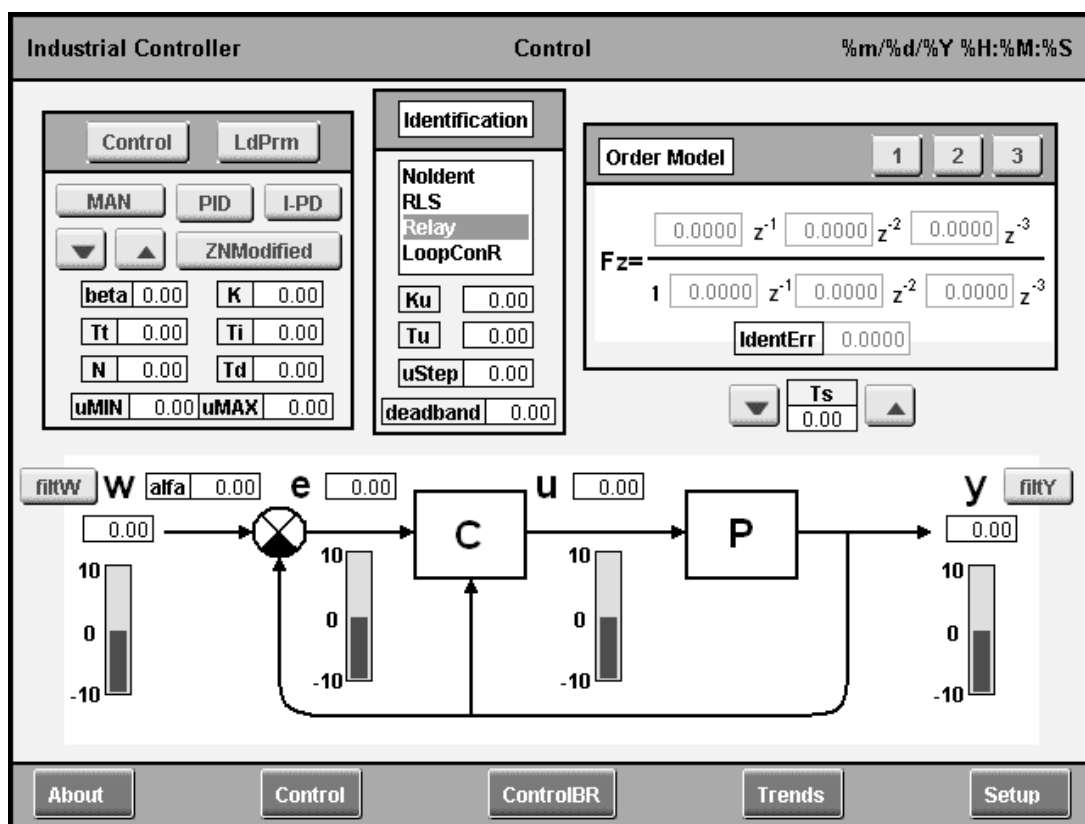
Navržený regulační člen je ovládán a vizualizován pomocí dotykové obrazovky, kterou PP400 obsahuje. Rozhraní bylo vytvořeno ve Visual Components Editoru (které je obsaženo v Automation Studiu), který je postaven na platformě Automation Runtime. Vizualizace obsahuje pět stránek. Tyto stránky mají společnou vrstvu, kterou tvoří záhlaví a zápatí (Obr. 6.1). Záhlaví obsahuje vlevo název aplikace, uprostřed se dynamicky mění název aktuálně zobrazené stránky a vpravo je datum a čas. Zápatí tvoří pět tlačítek pro přepínání mezi jednotlivými stránkami.

Stránka pro ovládání navrženého regulačního členu je na Obr. 6.1. Horní část stránky tvoří tři rámečky. Rámeček vlevo slouží pro ovládání bloku Control. V záhlaví tohoto rámečku vlevo je tlačítko, které povoluje nebo zakazuje řízení. Vpravo je tlačítko, které slouží pro načtení kritické periody kmitů a kritického zesílení pro účely spočítání konstant regulátoru. V hlavní části rámečku jsou tlačítka pro přepínání mezi řídicími algoritmy, ovládání manuálního akčního zásahu a tlačítko pro výběr, jestli se mají parametry počítat podle klasické nebo modifikované metody ZN. Je také možné ručně zadat všechny konstanty regulátoru.

Prostřední rámeček slouží k ovládání bloku Ident. V komponentě ListBox můžeme vybrat jakou metodou se má identifikovat (Obr. 4.4). Pod ní jsou pole pro zobrazení kritických parametrů identifikované soustavy. Uživatel má možnost také zvolit amplitudu jednotkového skoku pro identifikaci pomocí rekurzivní metody nejmenších čtverců. Pole deadband slouží pro zadání, kolik vzorkovacích period má být algoritmus ve stavu pásma necitlivosti po průchodu žádanou hodnotou, při identifikaci pomocí relé ve zpětné vazbě (4.1.2).

Rámeček vpravo má informativní charakter při identifikaci pomocí metody nejmenších čtverců. Zobrazují se zde parametry identifikovaného procesu. V záhlaví se volí řád modelu, který se má identifikovat. Pod rámečkem je možnost změny periody vzorkování.

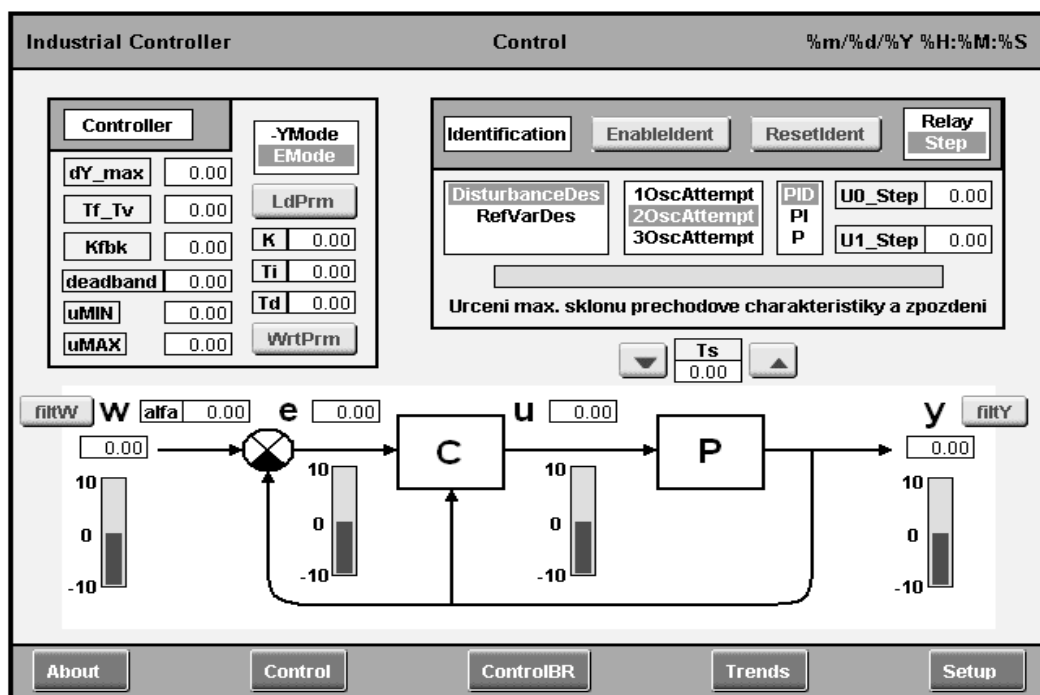
Spodní část stránky zobrazuje aktuální hodnoty procesních veličin buď přímo číselně nebo pomocí BarGraphu. Zadává se zde žádaná hodnota a jsou tu dvě tlačítka pro možnost zapnutí nebo vypnutí filtrů žádané hodnoty a regulované veličiny.



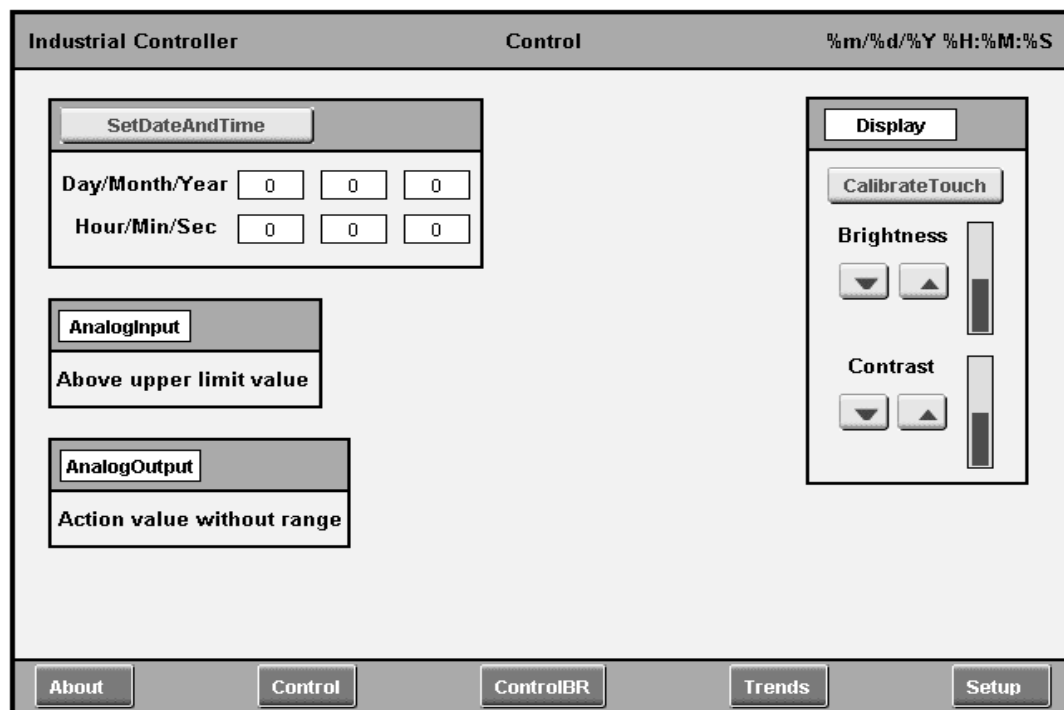
Obr. 6.1 Stránka vizualizace pro ovládání regulačního členu

Stránka pro ovládání a vizualizaci adaptivního regulátoru firmy B&R je na Obr. 6.2. Spodní část je stejná jako pro navržený regulační člen. V horní části jsou dva rámečky. Levý slouží pro nastavení PID regulátoru. Tlačítkem LdPrm se načítají parametry regulátoru určené identifikací. Tlačítkem WrtPrm se dají nastavit parametry regulátoru uživatelem.

Pomocí ovládacích prvků v pravém rámečku se nastavují parametry identifikace. Uživatel má možnost vybrat mezi identifikací pomocí skoku a relé ve zpětné vazbě. Pro identifikaci skokem si může uživatel vybrat mezi nastavením parametrů regulátoru na poruchu nebo žádanou hodnotu. U relé je možno zvolit kolik fází bude identifikace mít. Volí se zde i typ regulátoru. Ve spodní části rámečku se zobrazují informace o tom, v jakém stavu se právě nachází identifikace. Aby bylo možno řídit a identifikovat pomocí této obrazovky je nutné, aby na stránce Control (Obr. 6.1) byla v komponentě ListBox vybrána položka LoopConR.



Obr. 6.2 Stránka vizualizace pro adaptivní regulátor firmy B&R



Obr. 6.3 Stránka pro dodatečná nastavení

Na stránce s názvem Trends je možnost sledovat průběhy procesních veličin, tzn. žádané hodnoty, regulované veličiny a akčního zásahu.

Na stránce Setup (Obr. 6.3) se nastavuje datum a čas. Dále je zde tlačítko pro kalibraci dotykové obrazovky a tlačítka pro upravení kontrastu a jasu. Dva levé dolní rámečky dávají informaci o stavu analogového vstupu a analogového výstupu.

Ovládání a popis vizualizace je zde jen nastíněn. Detailnější popis funkčnosti může nalézt čtenář v příloze.

7. OVĚŘENÍ A SIMULACE

Správnost naprogramovaných algoritmů pro řízení a identifikaci byla ověřována v matlabu a simulinku. V simulinku byl vždy řídicí algoritmus naprogramován jako s-funkce a současně sestaven jako schéma v simulinku. Oba takto sestavené algoritmy musely dávat stejné odezvy na stejné vstupní veličiny. Identifikační algoritmus rekurzivní metody nejmenších čtverců se směrovým zapomínáním byl také nejprve testován jako s-funkce. V m-file s-funkci je naprogramování tohoto algoritmu jednodušší z důvodu podpory maticového počtu. Při implementaci v jazyce ANSI C už přímá podpora maticového počtu není.

Po ověření správnosti algoritmů bylo navržené řešení včetně vizualizace implementováno v Automation Studiu 3.0.71 v jazyce ANSI C. Testování probíhalo nejprve na simulátoru PLC AR000, které je součástí Automation Studia. Simulátor se chová jako stanice, na kterou je možno se připojit pomocí ethernetu. Z Automation Studia je proto možné do něj nahrát přeložený projekt. Simulátor má IP adresu 127.0.0.1.

K testování správnosti implementace v ANSI C bylo nejprve využíváno simulinku a v něm bloku mk_pvi. Pomocí tohoto bloku je možné se ze simulinku připojit k určitému tasku, vykonávanému na PLC přes IP adresu a název příslušného tasku. Tyto náležitosti musejí být nastaveny v souboru mk_pvi.ini, který musí být ve stejném adresáři jako modelovací schéma simulinku. Blok má jeden vstup a jeden výstup. Výstupem z bloku byl akční zásah, který sloužil jako vstup do matematického modelu řízené soustavy v simulinku. Vstupem do bloku mk_pvi byla hodnota regulované veličiny ze simulinku, která se v PLC zpracovávala. Další možností jak simulovat reálnou regulovanou soustavu matematickým modelem je FB LCRPT1 nebo LCRPT2, který je součástí knihovny LoopConR. Tyto bloky představují setrvačné články prvního a druhého řádu.

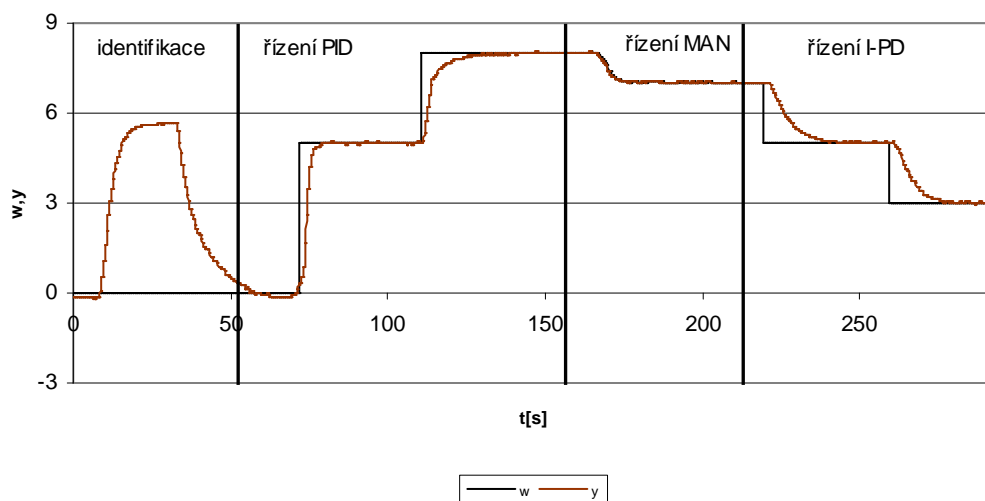
Testování přišlo na PP400 a reálném modelu v laboratoři, po odladění projektu na matematických modelech. PP400 (stanice DA09 v laboratoři s IP adresou 147.229.76.20) byl pomocí analogového vstupu (X20AI2622) a analogového výstupu (X20AO4632) spojen se vzduchovým tunelem. Jedním z akčních členů v

tomto modelu je větrák, který jako regulovanou veličinu řídí průtok vzduchu tunelem. Průtok vzduchu je měřen pomocí vrtulky.

Identifikace pomocí rekurzivní metody nejmenších čtverců byla nejprve implementovaná s obyčejným exponenciálním zapomínáním. Pokud se systém dostal do ustáleného stavu a koeficient exponenciálního zapomínání byl v tu dobu menší než jedna, prvky matice P s časem rostly. Při nastavení koeficientu zapomínání na jedničku k tomuto nedocházelo. Při implementaci směrového zapomínání i pro hodnoty koeficientu menší než jedna v ustáleném stavu nedocházelo u matice P k estimator windupu. Při testování algoritmu na některých matematických modelech, algoritmus nejprve identifikoval správné parametry, tzn. že kritické hodnoty spočítané pomocí těchto parametrů byly použitelné pro návrh regulátoru. Zároveň byla difference mezi skutečnou a predikovanou hodnotou téměř rovna nule. Po nějaké době se však parametry naráz změnily a tím se změnily i kritické parametry. Difference mezi odhadovaným a skutečným výstupem však byla pořád téměř nulová. Tento problém se objevoval při testování v PLC. Při testování v matlabu se toto nestalo. Proto byl předpoklad, že je to kvůli nedostatečnému rozlišení datového typu REAL v Automation Studiu. Použil jsem proto datový typ LREAL, ale problém se tím neodstranil. Stávalo se to především u soustav, jejichž diskrétní přenos měl při dané periodě vzorkování v čitateli koeficienty v řádu 10^{-5} . Pokud byl algoritmus testován na soustavě, jejíž diskrétní přenos měl koeficienty maximálně v řádu 10^{-2} tento problém nenastal. Z toho důvodu byl problém pravděpodobně v přesnosti a zaokrouhlování. Problém by mohl být možná vyřešen, pokud by byl implementován odmocninový algoritmus adaptace matice P .

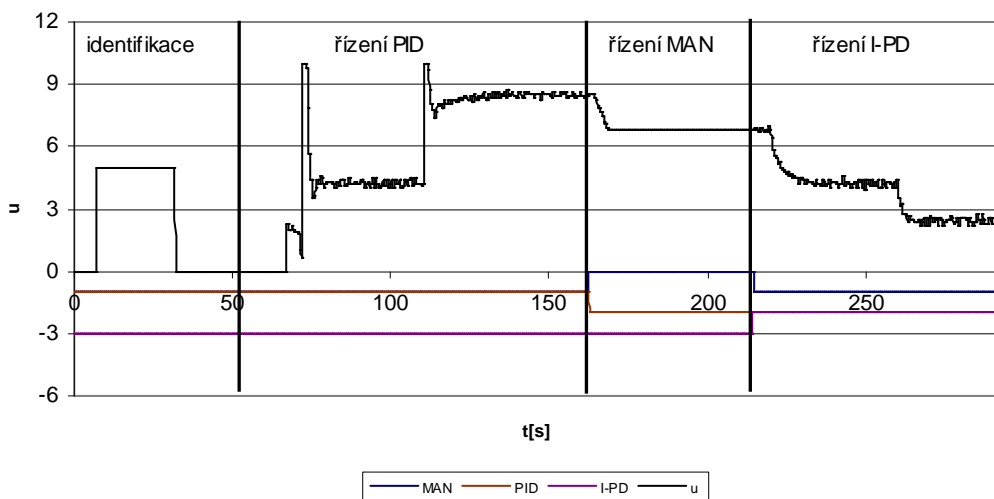
Průběhy při identifikaci pomocí rekurzivní metody nejmenších čtverců a následného řízení vzduchového tunelu je na Obr. 7.1 a Obr. 7.2. Nejprve se identifikovalo a poté se podle nalezených kritických parametrů spočítaly konstanty regulátoru podle modifikovaného ZN (odstranění překmitu regulované veličiny). Při nulové regulační odchylce došlo k přepnutí regulačních algoritmů. Z průběhu akčního zásahu je vidět, že přepnutí bylo beznárazové. Regulovaná veličina se ustálila na žádané hodnotě bez překmitu. Zjištěné kritické parametry a hodnoty konstant regulátoru jsou uvedeny v Tab. 7.1, kde T je vzorkovací perioda.

Průběh žádané a regulované veličiny při identifikaci RLS a následné řízení



Obr. 7.1 Průběh žádané a regulované veličiny při RLS

Průběh akčního zásahu při identifikaci RLS a následné řízení



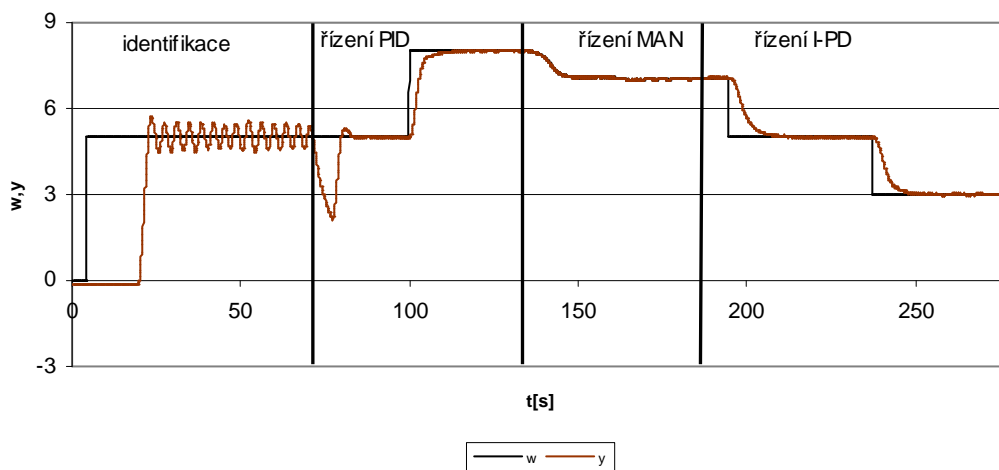
Obr. 7.2 Průběh akčního zásahu při RLS

$T = 0,5s$	$K = 2,42$	$N = 10$	$UMIN = 0$
$T_u = 5,15s$	$T_i = 2,42s$	$T_t = 8$	$UMAX = 10$
$K_u = 8,05$	$T_d = 0,64s$	$\beta = 0$	$uStep = 5$

Tab. 7.1 Konstanty regulátoru a soustavy při RLS

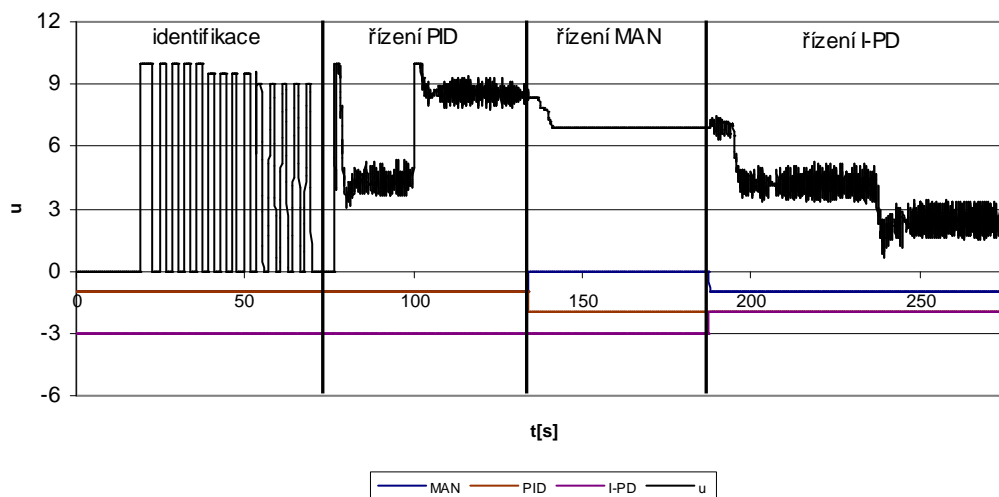
Průběhy veličin při identifikaci pomocí relé ve zpětné vazbě jsou na Obr. 7.3 a Obr. 7.4.

Průběh žádané a regulované veličiny při identifikaci pomocí relé a následné řízení



Obr. 7.3 Průběh žádané a regulované veličiny při identifikaci pomocí relé ve zpětné vazbě

Průběh akčního zásahu při identifikaci pomocí relé a následné řízení



Obr. 7.4 Průběh akčního zásahu při identifikaci pomocí relé ve zpětné vazbě

Z průběhu akčního zásahu při identifikaci je vidět, že identifikace skutečně probíhala ve třech fázích (4.1.2). Bylo opět přepnuto na všechny dostupné řídicí

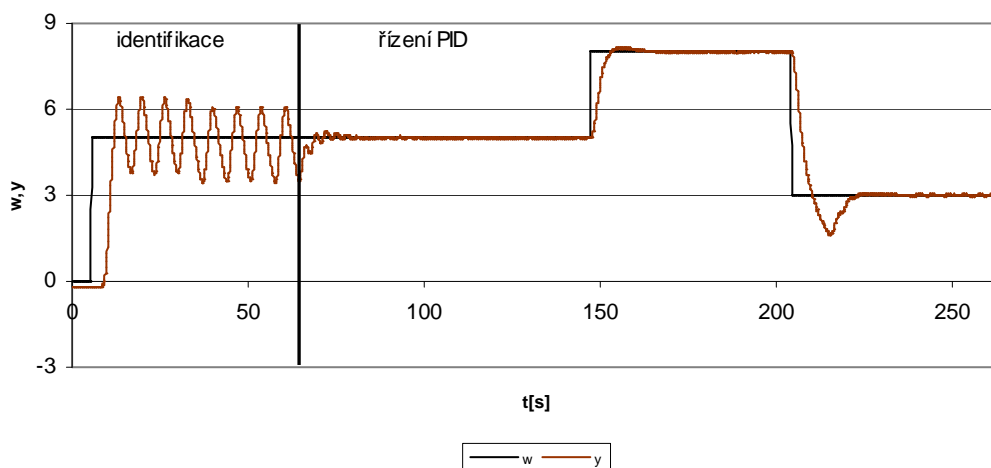
algoritmy pro ověření beznárazového přepnutí. Při přepnutí z manuálního režimu na I-PD se může zdát, že o beznárazové přepnutí nejde. Akční zásah je však při manuálním režimu nastaven přesně a při algoritmu I-PD se počítá podle aktuální regulované veličiny, která obsahuje šum. Zjištěné parametry a hodnoty konstant regulátoru pro relé jsou v Tab. 7.2. Konstanty byly počítány podle modifikovaného ZN. Akční zásah je oproti RLS více kmitavý. Je to způsobeno především větším zesílením v proporcionální a derivační části. Díky tomu se však regulovaná veličina ustálí na žádané hodnotě rychleji než u regulátoru nastaveného pomocí RLS.

$T = 0,1s$	$K = 3,59$	$N = 10$	$U_{MIN} = 0$
$T_u = 3,6s$	$T_i = 3,6s$	$T_t = 8$	$U_{MAX} = 10$
$K_u = 11,96$	$T_d = 0,45s$	$\beta = 0$	$w = 5$

Tab. 7.2 Konstanty regulátoru a soustavy při použití relé ve zpětné vazbě

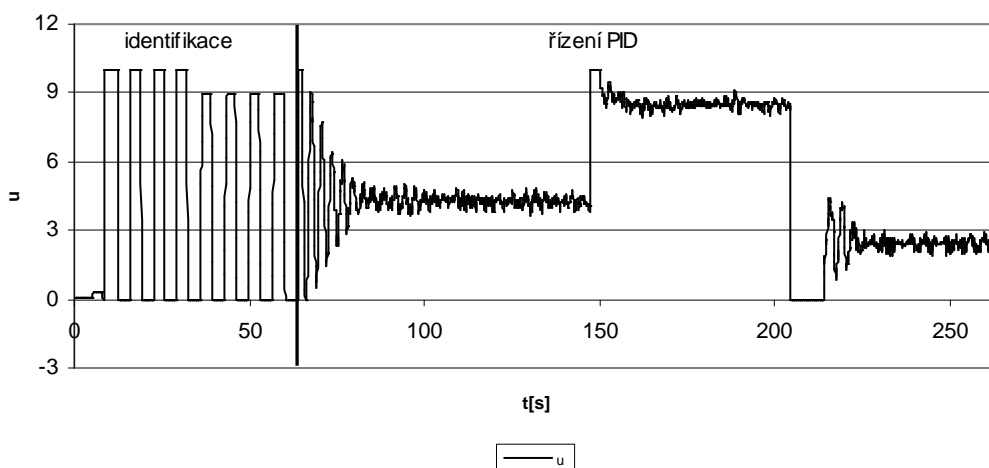
Průběhy při identifikaci pomocí relé od B&R jsou na Obr. 7.5 a Obr. 7.6. Pokud by byl zobrazen akční zásah a regulovaná veličina v jednom grafu, bylo by vidět, že je použito relé s hysterezí (relé se na výstupu nepřepne hned po průchodu žádanou hodnotou). Z toho důvodu má regulovaná veličina při identifikaci větší

Průběh žádané a regulované veličiny při identifikaci pomocí relé LoopConR a následné řízení



Obr. 7.5 Průběh žádané a regulované veličiny při identifikaci pomocí relé ve zpětné vazbě pomocí LoopConR

Průběh akčního zásahu při identifikaci pomocí relé LoopConR a následné řízení



Obr. 7.6 Průběh akčního zásahu při identifikaci pomocí relé ve zpětné vazbě pomocí LoopConR

amplitudu. Regulovaná veličina se ustálila na žádané hodnotě s překmitem. Hodnoty konstant regulátoru jsou v Tab. 7.3. Pro identifikaci byly zvoleny tři fáze, ale identifikace skončila po druhé fázi. Důvodem je to, že zjištěné parametry již jsou dostatečně přesné (přesnost se dá nastavit).

$T = 0,1s$	$K = 3,45$	$w = 5$
$U_{MIN} = 0$	$T_i = 0,86s$	
$U_{MAX} = 10$	$T_d = 0,13s$	

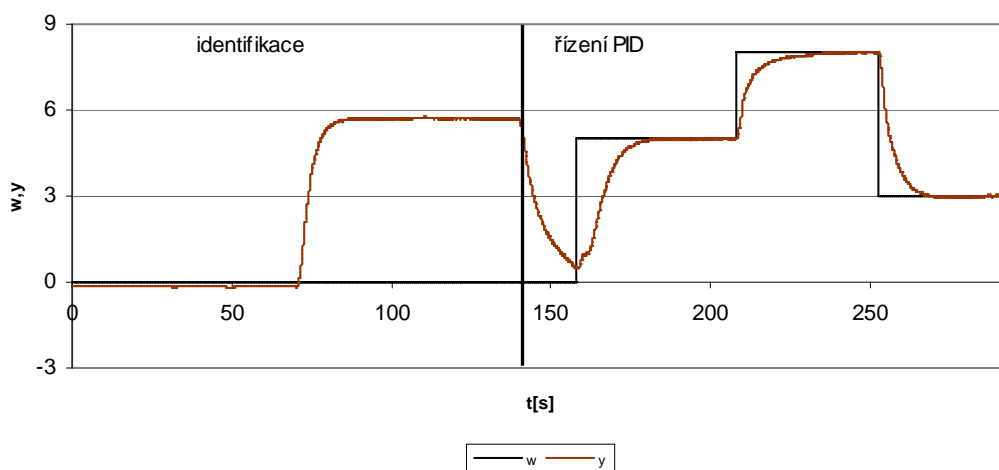
Tab. 7.3 Konstanty regulátoru při použití relé ve zpětné vazbě pomocí LoopConR

Druhou možností jak identifikovat pomocí LoopConR je odezva na skok. Průběhy veličin jsou na Obr. 7.7 a Obr. 7.8. Regulátor se nastavoval na sledování žádané hodnoty, což je vidět z průběhu identifikace (4.1.3). Regulovaná veličina se ustálila oproti identifikaci pomocí relé na žádané hodnotě bez překmitu. To je ale vykoupeno pomalejší odezvou. Konstanty regulátoru jsou v Tab. 7.4

$T = 0,1s$	$K = 5,13$
$U_{MIN} = 0$	$T_i = 0,9s$
$U_{MAX} = 10$	$T_d = 0,13s$

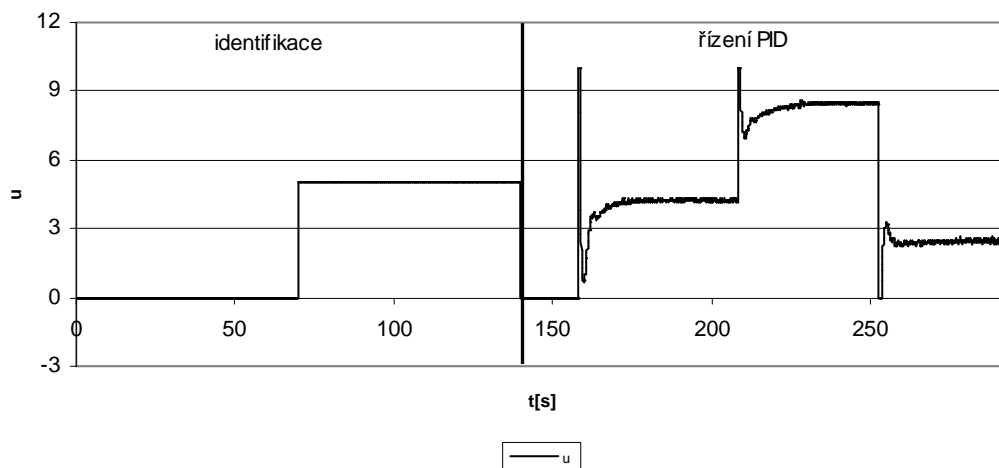
Tab. 7.4 Konstanty regulátoru při identifikaci skokem pomocí LoopConR

Průběh žádané a regulované veličiny při identifikaci pomocí skoku na odezvu na žádanou hodnotu LoopConR a následné řízení



Obr. 7.7 Průběh žádané a regulované veličiny při identifikaci z jednotkové odezvy pomocí LoopConR

Průběh akčního zásahu při identifikaci pomocí skoku na odezvu na žádanou hodnotu LoopConR a následné řízení



Obr. 7.8 Průběh akčního zásahu při identifikaci z jednotkové odezvy pomocí LoopConR

Navržený regulační člen má možnost identifikace pomocí RLS a relé ve zpětné vazbě. Adaptivní regulátor firmy B&R identifikuje rozkmitáváním a z odezvy na jednotkový skok. U identifikace je možné nastavit mnohem více parametrů než

v navrženém regulačním členu. Bloky mají také stavové výstupy, které informují o tom, v jakém stavu se zrovna identifikace nachází. U B&R je možno zvolit mezi P, PI a PID regulátorem. V navrženém regulačním členu je možný PID a I-PD algoritmus. U B&R regulátoru se dá nastavit více parametrů u samotné struktury regulátoru.

8. ZÁVĚR

V úvodní části zpracování projektu byla ověřena správnost naprogramování jednotlivých dílčích algoritmů pro řízení a identifikaci v prostředí Matlab/Simulink. Algoritmy byly naprogramovány jako s-funkce a porovnávány se schémata jednotlivých algoritmů složených z bloků.

Ve druhé části byl proveden strukturovaný návrh softwaru [9]. Celý problém byl rozdělen na dílčí podproblémy a bylo stanoveno rozhraní mezi těmito podproblémy. Pro každou dílčí část výsledného řešení byl sestaven stavový automat nebo vývojový diagram. Následovala implementace jednotlivých částí v ANSI C jako tasků v prostředí Automation Studio verze 3.0.71. Díky tomuto návrhu byla tvorba softwaru jednodušší, program je čitelnější a snadněji se udržuje.

Navržený regulační člen má dvě možnosti identifikace. Pomocí relé ve zpětné vazbě a pomocí rekurzivní metody nejmenších čtverců se směrovým zapomínáním. Identifikace pomocí relé ve zpětné vazbě má tři fáze. Je použito ideální a saturační relé [6] kvůli zpřesnění hledaných kritických parametrů řízené soustavy. Při ověřování na matematickém modelu, kde byly známy přesné hodnoty kritických parametrů, se toto potvrdilo. Kritické parametry spočítané po třetí fázi byly přesnější, než kritické parametry spočítané po druhé fázi. Nevýhoda tohoto přístupu je u pomalejších soustav, protože identifikace trvá déle. Výstupem tohoto typu identifikace je kritická perioda kmitů a kritické zesílení.

Rekurzivní metoda nejmenších čtverců se směrovým zapomínáním [3] odstraňuje tzv. estimator windup [2]. K tomuto jevu dochází při obyčejné RLS s exponenciálním zapomínáním, pokud je systém v ustáleném stavu. Algoritmus identifikuje první, druhý nebo třetí řád, podle volby uživatele. Z parametrů modelu je nutné určit kritické hodnoty řízené soustavy, které jsou potřebné pro návrh konstant regulátoru. Parametry jsou počítány bez rozkmitávání regulované soustavy [4]. Takto získané kritické parametry jsou někdy pro řízení jen těžko použitelné. Parametry modelu jsou pomocí RLS nejprve určeny přesně, ale po delším čase se náhle skokově změní. Metoda sice dále odhaduje výstup modelu s malou chybou, ale parametry modelu neodpovídají skutečným hodnotám. Proto navržené konstanty regulátoru

nejdou pro řízení vhodné. Řešením by mohlo být použití odmocninového algoritmu pro adaptaci kovarianční matice P .

Řízení je možno realizovat ručně nebo pomocí jednoho ze dvou řídicích algoritmů, diskrétního PID regulátoru s filtrací derivační složky a antiwindupem a I-PD regulátoru. Stavový automat pro blok řízení je navržen tak, aby bylo možné ho bez problémů rozšířit o další řídicí algoritmy. Mezi manuálním řízením a řídicími algoritmy je možno beznárazově přepínat, což je požadavkem u průmyslových regulátorů. Konstanty regulátorů může uživatel nastavit sám nebo mohou být spočítány pomocí kritických hodnot regulované soustavy podle klasických nebo modifikovaných (omezení překmitu regulované veličiny) pravidel metody Zieglera-Nicholse.

Pro porovnání vytvořeného regulačního členu byl implementován adaptivní regulátor firmy B&R jako funkční blok LCRSlimPID() z knihovny LoopConR. Adaptivní regulátor má dvě možnosti identifikace: rozkmitáváním a z odezvy na jednotkový skok akčního zásahu. U obou metod je možné nastavit spoustu parametrů, aby identifikace vyhovovala regulované soustavě. U rozkmitávání se volí např. kolik fází bude identifikace mít. U identifikace z odezvy na jednotkový skok může uživatel zvolit návrh konstant regulátoru na poruchu nebo na sledování žádané hodnoty. Blok má stavový výstup pomocí něhož můžeme sledovat, v jaké fázi se právě identifikace nachází. U regulátoru je možno zohlednit akční člen nejen maximální a minimální hodnotou, ale i maximálním nárůstem akčního zásahu. Regulátor může být typu P, PI nebo PID.

Byla vytvořena vizualizace pro dotykovou obrazovku ve Visual Components Editoru, který je součástí Automation Studia. Tvoří ji pět stránek, mezi kterými se může uživatel přepínat pomocí tlačítek. Ovládací prvky jsou přehledně umístěny do rámečků. Na stránkách určených pro ovládání uživatel zvolí vše potřebné pro řízení nebo identifikaci a hned vidí číselně a pomocí BarGraphu hodnoty procesních veličin. Pro detailnější analýzu průběhů se může přepnout na obrazovku trendů, kde jsou zobrazeny žádaná hodnota, regulovaná veličina a akční zásah. Vizualizace také obsahuje stránku Setup. Ta slouží pro nastavení obrazovky a obsahuje informaci o stavu analogového vstupu a výstupu.

Třetí částí bylo ověřování správnosti a simulace vytvořeného regulátoru. Ta byla nejprve prováděna spojením simulátoru PLC a simulinku pomocí bloku mk_pvi. Jako řízená soustava sloužil model v simulinku. Jako matematický model řízené soustavy byl také použit funkční blok z knihovny LoopConR. To znamená, že regulovanou soustavu představoval přímo task ve virtuálním PLC. Po odladění byl regulátor ověřován na Power Panelu 400 v laboratoři. Ten byl spojen pomocí analogového výstupu a analogového vstupu se vzduchovým tunelem. Akční člen zde byl větrák, který řídil průtok vzduchu v tunelu. Na tunelu byly odzkoušeny všechny metody identifikace a možnosti řízení.

Seznam literatury

- [1] PIVOŇKA, P.: *Optimalizace regulátorů*. Skriptum. VUT, Brno, 2005.
- [2] ÅSTRÖM, K. J., WITTENMARK, B.: *Adaptive Control*. 2nd edition. Reading, Massachusetts : Addison-Wesley, 1995. 574 s. ISBN 0-201-55866-1.
- [3] CAO, L., SCHWARTZ, H. M.: *A directional forgetting algorithm based on the decomposition of information matrix*. Proceedings of 7th Mediterranean Conference on Control and Automation. 1999, s. 1635-1644.
- [4] BOBÁL, V. a kol.: *Praktické aspekty samočinně se nastavujících regulátorů a implementace*. VUT Brno, 1999.
- [5] HANG, C. C., ASTROM, K. J., WANG, Q. G.: *Relay feedback auto-tuning of process controllers – a tutorial review*. In Journal of Process Control. 2002th edition. 2002. Vol. 12. s. 143-162.
- [6] YU, Cheng-Ching.: *Autotuning of PID Controllers : A Relay Feedback Approach*. 2nd Edition. [s.l.] : Springer, 2006. 261 s. ISBN 1-84628-036-2
- [7] PIVOŇKA, P., SCHMIDT, M.: *Comparative Analysis of Discrete Derivative Implementations in PID Controllers*. In Systems Theory and Applications Greece 2007, pp. 33-37.
- [8] PIVOŇKA, P.: *Comparative Analysis in Implementations Discrete PID Controllers*.
- [9] B&R AUTOMATION.: *Structured Software Generation*. TM230,
<Q:/VYUKA/CHOMAT/B&R/ŠkolicíMateriály_AS3/Controls/AS_C_TR2/TM230TRE.30-ENG.pdf>
- [10] B&R AUTOMATION.: *Automation Studio 3.0.71 – Online Help*
- [11] PIVOŇKA, P.: *Číslicová řídicí technika*. Skriptum. VUT, Brno, 2004.
- [12] PRUDKÝ, M. *Průmyslový regulátor PID s autotunerem a vizualizací*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 96 s. Vedoucí diplomové práce prof. Ing. Petr Pivoňka, CSc.
- [13] B&R AUTOMATION.: *Automation Runtime*. TM213,
<Q:/VYUKA/CHOMAT/B&R/ŠkolicíMateriály_AS3/Controls/AS_C_TR1/TM213TRE.25-ENG.pdf>

Seznam zkratek

ANSI C	standard programovacího jazyka C publikovaný American National Standards Institute
AR000	virtuální PLC firmy B&R
B&R	zkratka rakouské firmy Bernecker + Rainer Industrie-Elektronik Ges.m.b.H.
D/A	digitálně/analogový
FB	funkční blok
I/O	vstupně výstupní
INT	datový typ integer
IP	IP adresa
I-PD	I-PD regulátor
LREAL	datový typ long double
mA	jednotka elektrického proudu miliampér
ms	jednotka času milisekunda
Obr.	obrázek
P	P regulátor
PI	PI regulátor
PID	PID regulátor
PLC	programovatelný automat
PP400	Power Panel 400
REAL	datový typ double
RLS	rekurzivní metoda nejmenších čtverců
Tab.	tabulka
tzn.	to znamená
tzv.	takzvaně
V	jednotka elektrického napětí volty
ZN	Ziegler-Nichols

Seznam symbolů

a	amplituda odchylky
\bar{a}	maximální hodnota vstupu saturačního relé, při které je výstup přímo úměrný vstupu
B_n	koeficient Fourierovy transformace
d	počet kroků dopravního zpoždění
$D(z)$	charakteristický polynom
dif	stav difference PID regulátoru
e	regulační odchylka
$E(z)$	obraz regulační odchylky
$F_D(z)$	obraz přenosu derivační větve
$G(j\omega)$	frekvenční přenos soustavy
$G_P(z)$	přenos soustavy
$G_R(z)$	přenos regulátoru
h	amplituda relé
$h(k)$	diskrétní odezva na jednotkový skok
K	zesílení proporcionální části regulátoru
k	sklon saturačního relé
k_{min}	minimální sklon saturačního relé
$K_P(z)$	obraz proporcionálního zesílení
K_{PK}	kritické zesílení
K_s	statické zesílení
K_u	kritické zesílení
M	matice M
N	konstanta omezující zesílení rušivých signálů na vyšších frekvencích (volí se v rozmezí 3 – 20)
$N(a)$	náhrada relé Fourierovou transformací
P	kovarianční matice P
R	informační matice R
sum	stav sumace PID regulátoru
t	čas

T	vzorkovací perioda
T_1	časová konstanta filtru žádané hodnoty
T_D	derivační konstanta regulátoru
T_I	integrační konstanta regulátoru
T_k	kritická perioda kmitů
T_T	sledovací konstanta antiwindupu
T_u	kritická perioda kmitů
u	akční zásah
$U(z)$	obraz akčního zásahu
u_{INT}	akční zásah datového typu integer
U_{MAX}	maximální akční zásah
U_{MIN}	minimální akční zásah
u_{real}	akční zásah datového typu real
u_{Step}	amplituda jednotkového skoku při identifikaci pomocí RLS
$u_{stř.}$	střední hodnota akčního zásahu
w	žádaná hodnota
$W(z)$	obraz žádané hodnoty
w^*	nefiltrovaná žádaná hodnota
x	regulovaná veličina
y	regulovaná veličina
$Y(z)$	obraz regulované veličiny
y_{INT}	regulovaná veličina datového typu integer
\hat{y}	odhadovaný výstup
y_{real}	regulovaná veličina datového typu real
z	kladný operátor Z-transformace
z^{-1}	záporný operátor Z-transformace
α	koeficient filtru žádané hodnoty
β	koeficient zesílené žádané hodnoty u I-PD regulátoru
ϵ	chyba mezi skutečnou a odhadovanou výstupní veličinou
Φ	matice minulých vstupů a výstupů

γ	úhel charakterizující saturační relé
φ	regresní vektor
λ	koeficient exponenciálního zapomínání
θ	vektor odhadovaných parametrů
ω	úhlová frekvence
ω_k	kritická úhlová frekvence
ω_u	kritická úhlová frekvence

Seznam příloh

Průmyslový PID regulátor s autotunerem a vizualizací – popis a návod

Příloha č.1

Průmyslový PID regulátor s autotunerem a vizualizací – popis a návod

Bc. Pavel Vávra

1. OBECNÝ POPIS REGULAČNÍHO ČLENU

Navržený regulační člen má možnost identifikace pomocí dvou metod. Rekurzivní metody nejmenších čtverců a pomocí relé ve zpětné vazbě. Řízení je možné pomocí dvou řídicích algoritmů. Diskrétním PID regulátorem s filtrací derivační složky a dynamickým antiwindupem a I-PD regulátorem. Proces lze řídit i ručně. Mezi řídicími režimy je možno se beznárazově přepínat. Konstanty regulátoru se počítají pomocí pravidel Zieglera-Nicholse ze získaných kritických parametrů identifikované soustavy. Uživatel je samozřejmě může nastavit sám na základě zkušeností.

Identifikace a řízení je dále možné pomocí adaptivního regulátoru firmy B&R. Identifikuje se z přechodové charakteristiky nebo rozkmitáváním. Pro řízení může uživatel volit mezi P, PI a PID regulátorem.

Regulátor je ovládán pomocí vizualizace přes dotykovou obrazovku. Vizualizace má pět stránek. Úvodní stránku, stránku pro ovládání navrženého regulátoru, stránku pro ovládání adaptivního regulátoru firmy B&R, stránku s časovými průběhy procesních veličin a stránku pro nastavení dotykové obrazovky.

2. POUŽITÝ HARDWARE

Regulátor byl implementován do zařízení Power Panel 400 (PP400) na pracovišti DA09 v laboratoři E541. PP400 měl IP adresu 147.229.76.20. Hardwarová konfigurace je na Obr. 1. K PP400 byl připojen vzduchový tunel. Jako akční člen byl použit větrák, který byl ovládán druhým analogovým výstupem na kartě X20AO4632. Regulovaná veličina (proud vzduchu) byla snímána pomocí měřicí vrtulky. Hodnota byla měřena na druhém vstupu druhé vstupní analogové karty X20AI2622.

4PP480.1043-75	X20BR9300	X20AO4632	X20AI2622	X20AI2622	X20DO9322	X20DI9371
----------------	-----------	-----------	-----------	-----------	-----------	-----------

Obr. 1 Hardwarová konfigurace PLC

Po nahrání je potřeba provést warm restart, aby se provedly inicializační části jednotlivých tasků. Jinak program zhavaruje v důsledku dělení nulou.

3. OVLÁDÁNÍ A VIZUALIZACE

Vizualizace je tvořena pěti stránkami. Stránky mají společnou vrstvu, kterou tvoří záhlaví a zápatí. Záhlaví obsahuje vlevo název aplikace, uprostřed se dynamicky mění název aktuálně zobrazené stránky a vpravo je datum a čas. Zápatí tvoří pět tlačítek pro přepínání mezi jednotlivými stránkami. Stránka About (Obr. 2) má pouze informativní charakter. Dotykem kamkoli na obrazovku se přepneme na ovládací stránku regulátoru.

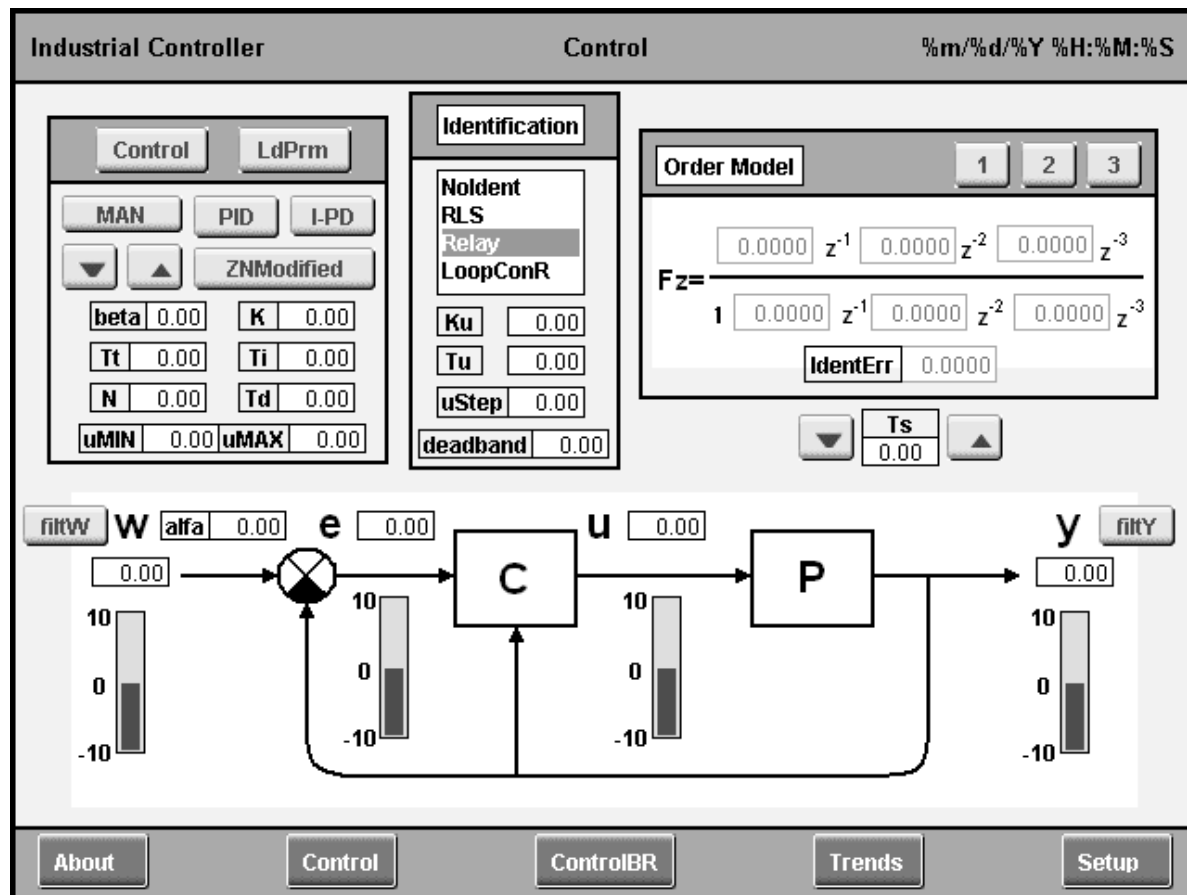


Obr. 2 Stránka About

3.1 OVLÁDÁNÍ NAVRŽENÉHO REGULAČNÍHO ČLENU

Stránka pro ovládání navrženého regulačního členu je na Obr. 3. Horní část stránky tvoří tři rámečky. Rámeček vlevo slouží pro ovládání regulačních algoritmů. V záhlaví tohoto rámečku vlevo je tlačítko, které povoluje nebo zakazuje řízení. Vpravo je tlačítko, které slouží pro načtení kritické periody kmitů a kritického zesílení pro účely spočítání konstant regulátoru. V hlavní části rámečku jsou tlačítka pro přepínání mezi řídicími algoritmy a ručním režimem. Tlačítka s šipkami nahoru a dolů pod tlačítkem MAN slouží pro ruční zvyšování nebo snižování akčního zásahu. Vedle se nachází tlačítko pro volbu, jestli se mají konstanty regulátoru z kritických parametrů soustavy počítat pomocí klasických pravidel ZN nebo podle pravidel pro omezení překmitu. Ve zbývajících částech rámečku jsou vstupní číselná

pole pro ruční nastavení konstant regulátoru a maximální a minimální hodnoty akčního zásahu, který dokáže ještě akční člen zpracovat.



Obr. 3 Stránka vizualizace pro ovládání regulačního členu

Prostřední rámeček slouží k ovládání identifikace. V komponentě ListBox můžeme vybrat jakou metodou se má identifikovat. NoIdent je pro vypnutí identifikace, RLS pro rekurzivní metodu nejmenších čtverců, Relay pro identifikaci pomocí relé ve zpětné vazbě a LoopConR pro řízení a identifikaci pomocí adaptivního regulátoru od B&R. Pole Ku a Tu zobrazují kritické parametry identifikované soustavy. Uživatel má možnost také zvolit amplitudu jednotkového skoku pro identifikaci pomocí RLS (pole uStep). Vstupní pole deadband slouží pro zadání počtu vzorkovacích period, po který má být algoritmus identifikace pomocí relé ve zpětné vazbě ve stavu pásma necitlivosti po průchodu žádanou hodnotou (kvůli šumu). Při identifikaci pomocí relé je vždy dobré před započítím procedury zvolit žádanou hodnotu mezi maximální a minimální hodnotou akčního zásahu. Kolem této žádané hodnoty bude poté regulovaná veličina kmitat.

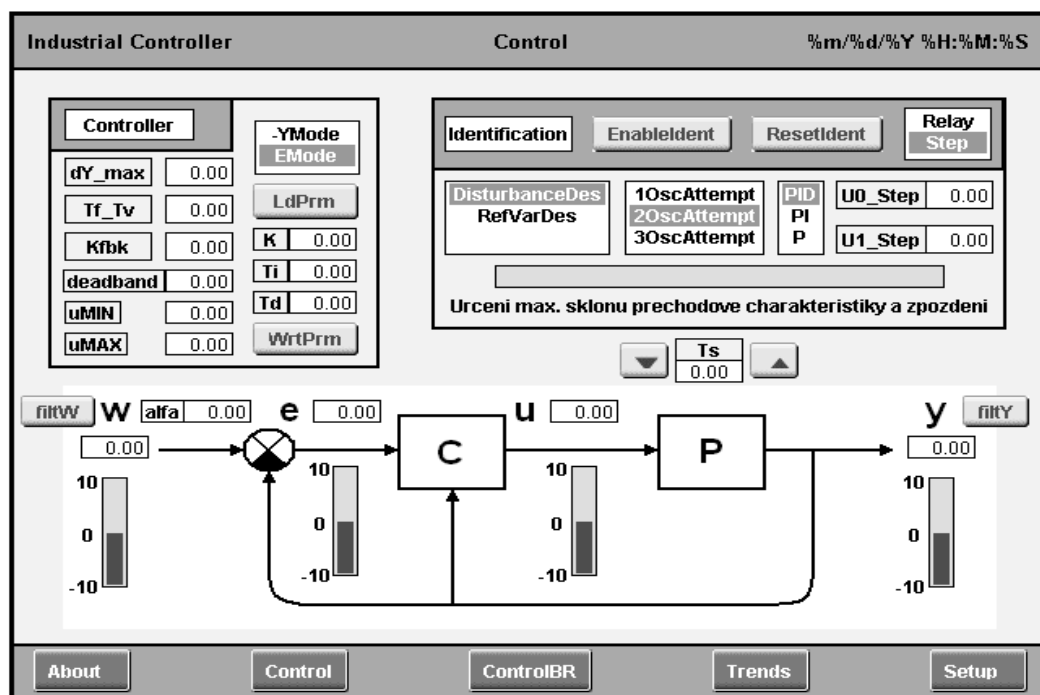
Rámeček vpravo má informativní charakter při identifikaci pomocí metody nejmenších čtverců. Zobrazují se zde parametry identifikovaného procesu. V záhlaví se volí řád modelu, který se má identifikovat. Pod rámečkem je možnost měnit periodu vzorkování.

Spodní část stránky zobrazuje aktuální hodnoty procesních veličin buď přímo číselně, nebo pomocí BarGraphu. Zadává se zde žádaná hodnota a jsou tu dvě tlačítka pro možnost zapnutí nebo vypnutí filtrů žádané hodnoty a regulované veličiny. Vstupní políčko alfa nastavuje koeficient α ve filtru žádané hodnoty.

3.2 OVLÁDÁNÍ ADAPTIVNÍHO REGULÁTORU FIRMY B&R

Stránka pro ovládání adaptivního regulátoru firmy B&R je na Obr. 4. Spodní část je stejná jako pro navržený regulační člen. V horní části jsou dva rámečky. Levý slouží pro nastavení PID regulátoru. Tlačítkem LdPrm se načítají parametry regulátoru určené identifikací. Tlačítkem WrtPrm se dají nastavit konstanty regulátoru uživatelem. Komponenta ListBox s položkami –YMode a EMode vybírá, jestli bude do derivační větve regulátoru vstupovat záporná regulovaná veličina nebo odchylka. Pomocí vstupních polí uMIN a uMAX se nastavuje maximální a minimální hodnota akčního zásahu. Do pole deadband může uživatel zadat pásmo necitlivosti pro regulační odchylku. Pokud bude regulovaná veličina v pásmu $\langle w - \text{deadband}; w + \text{deadband} \rangle$ bude regulační odchylka považována za nulovou. Kfbk je sledovací konstanta pro antiwindup. Tf_Tv určuje v jakém poměru bude filtrační časová konstanta derivační složky k derivační konstantě. A nakonec pole dY_max nastavuje maximální změnu akčního zásahu.

Pomocí ovládacích prvků v pravém rámečku se nastavují parametry identifikace. Uživatel má možnost vybrat mezi identifikací pomocí skoku a relé ve zpětné vazbě (ListBox s položkami Relay a Step). Pro identifikaci skokem si může uživatel vybrat mezi nastavením konstant regulátoru na poruchu nebo žádanou hodnotu (ListBox s položkami DisturbanceDes a RefVarDes). U relé je možno zvolit kolik fází bude identifikace mít (ListBox s položkami 1, 2 nebo 3OscAttempt). Volí se zde i typ regulátoru. Pro identifikaci pomocí přechodové charakteristiky se musí zadat akční zásah před skokem (U0_Step), který dostane regulovanou veličinu na úroveň žádané hodnoty a akční zásah po skoku (U1_Step). Ve spodní části rámečku se zobrazují informace o tom, v jakém stavu se právě nachází identifikace pomocí BarGraphu a popisem. Aby bylo možno řídit a identifikovat pomocí této obrazovky je nutné na stránce pro ovládání regulačního členu (Obr. 3) nastavit v komponentě ListBox položku LoopConR. Tlačítko EnableIdent spouští proces identifikace a tlačítko ResetIdent proces identifikace resetuje.



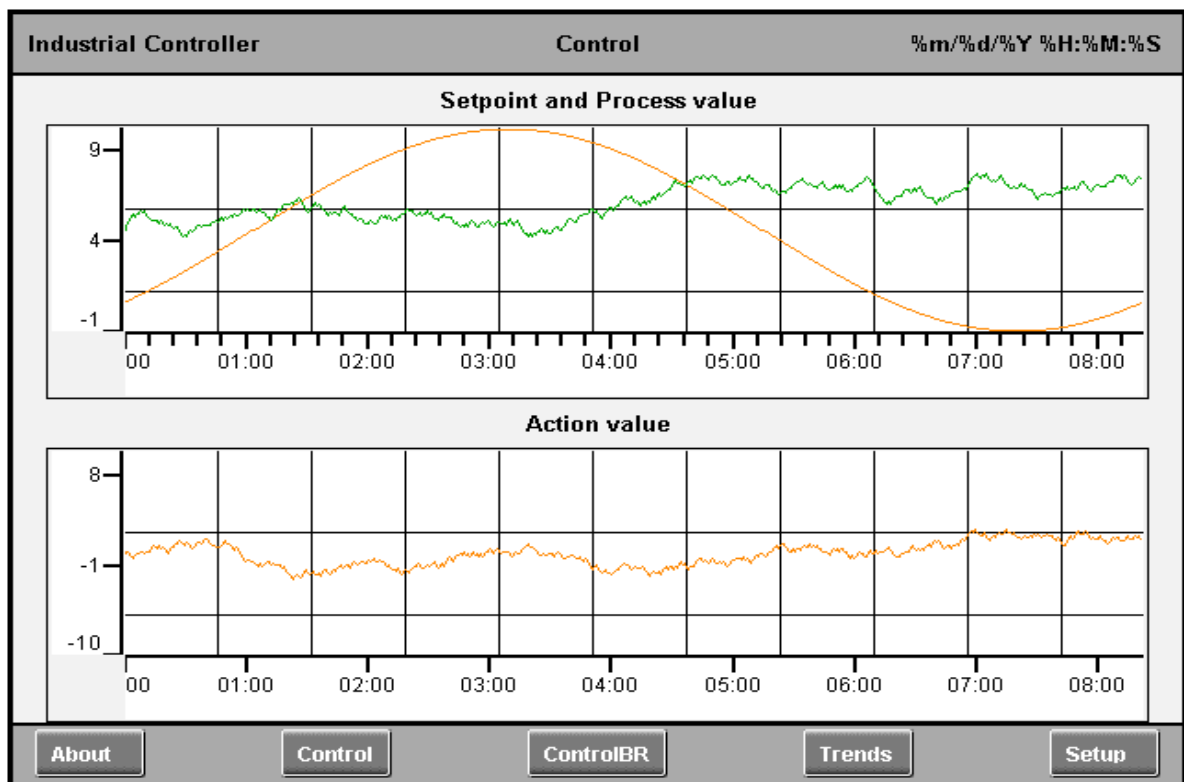
Obr. 4 Stránka vizualizace pro adaptivní regulátor firmy B&R

3.3 ZOBRAZENÍ PROCESNÍCH VELIČIN

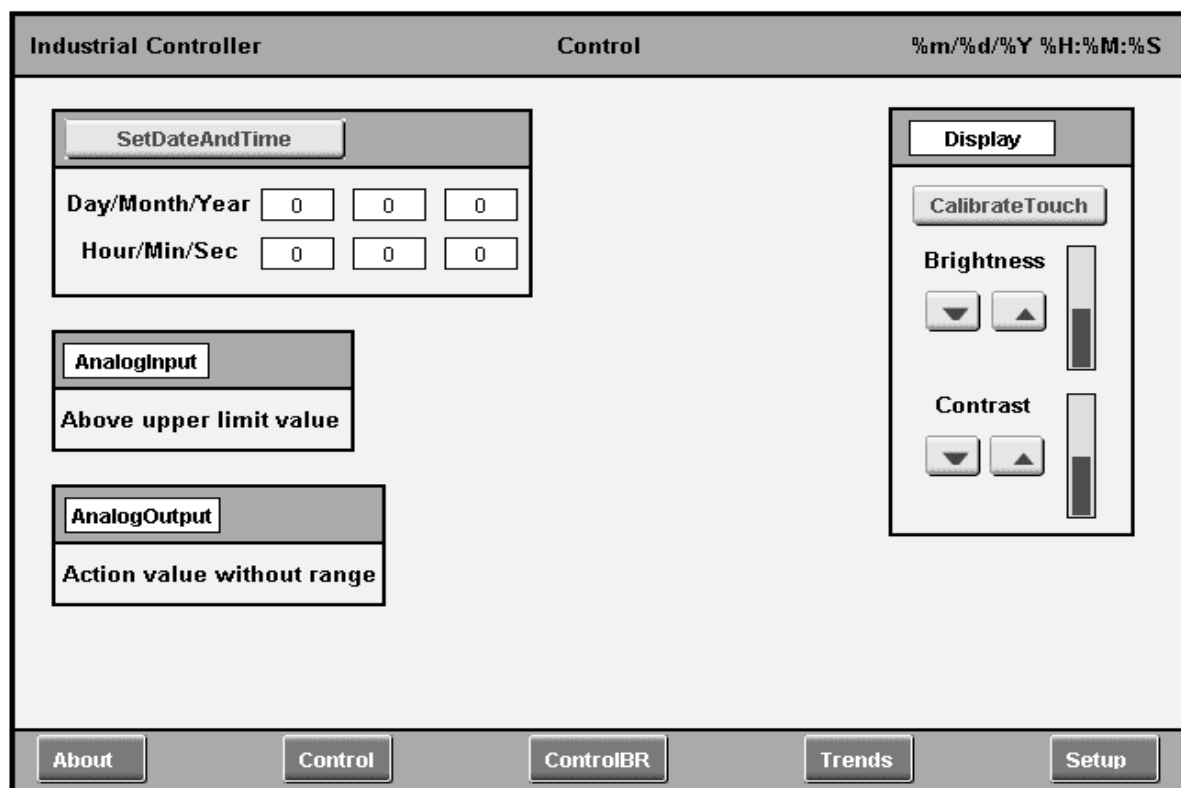
Stránka pro zobrazení procesních veličin je na Obr. 5. Nahoře je graf pro žádanou hodnotu a regulovanou veličinu. Spodní graf zobrazuje časový průběh akčního zásahu. Vyobrazené průběhy nejsou skutečnými průběhy, mají pouze informativní charakter

3.4 DODATEČNÁ NASTAVENÍ

Stránka s názvem Setup je na Obr. 6. Dají se zde nastavit kontrast a jas obrazovky tlačítky více méně. Aktuální hodnota je zobrazena pomocí BarGraphu. Tlačítko CalibrateTouch slouží ke kalibraci dotykové obrazovky. Pokud uživatel toto tlačítko použije, začnou se zobrazovat body. Těch se uživatel postupně dotýká a tím dojde ke kalibraci. Dále je možnost nastavit datum a čas pomocí tlačítka SetDateAndTime. Dva levé dolní rámečky dávají informaci o stavu analogového vstupu a analogového výstupu. Analogový vstup může mít čtyři stavy. V pořádku, měřená hodnota je mimo měřený rozsah (nad 10 V nebo pod -10 V) a signalizace přerušení vodiče. Analogový výstup může být ve dvou stavech. V pořádku nebo akční zásah mimo rozsah určený proměnnými uMIN a uMAX.



Obr. 5 Zobrazení procesních veličin



Obr. 6 Stránka pro dodatečná nastavení